1992年1月1日発行（毎月1回1日発行）第11巻1号通巻117号　昭和58年11月2日第三種郵便物認可

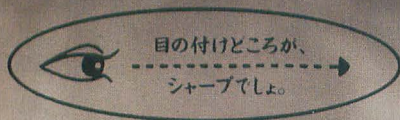
PERSONAL COMPUTER MAGAZINE for MZ, X1, and X68000

# 特集 SX-WINDOWの未来

ウィンドウシステムの比較/SX-WINDOWを検証する
MAGIC用3D迷路ゲーム/S-OSパズルゲーム「LINER」
製品紹介 Roland音源モジュールCM-300/500
第1回全日本X68000芸術祭予選レポート

1
1992

SOFT BANK
オー！エックス
定価600円

SHARP

# アプリも使うけど オリジナルツールも 創りたい。

X68000の世界に、思いきって踏み込んでみてください。アプリケーションの達人、ステーショナリーとしてのパソコン、それはそれで全く異論はないのですが、もっと新鮮な感動、驚き、発見に出会うはずです。コンピュータが本来持つ創造性、それとあなたの感性との接点に新しい何かが生まれる。グラフィック、サウンド＆ミュージック、エンターテイメント、X68000はさまざまなフィールドで、あなたの才能に応えるクリエイティブ環境を備えています。●クロック周波数16MHzの68000搭載 ●ウィンドウアプリケーションも続々登場、操作性を一段と高めたSX-WINDOW Ver.1.1搭載●メインメモリは標準で2MB、本体内に最大8MB、I/Oスロットを使えば最大12MBまで増設可能、数値演算プロセッサも本体内に取りつけ可能な高密度メモリ環境●大容量メディア対応、SCSIインターフェイス標準装備●X68000シリーズとフルコンパチブル設計。

## 瞬速16MHz、エクシヴィ快走。

エクシヴィ

X68エクシヴィ
▼
16 MHz

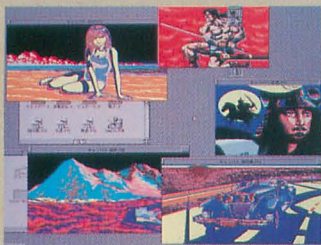
本体＋キーボード＋マウス・トラックボール
CZ-634C-TN(チタンブラック) 標準価格368,000円(税別)
81MB HDタイプ CZ-644C-TN(チタンブラック) 標準価格518,000円(税別)

●写真はCZ-644C-TNと別売の15型カラーディスプレイテレビCZ-614D-TN標準価格135,000円(税別)

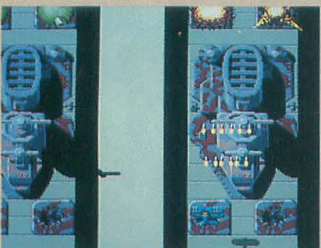
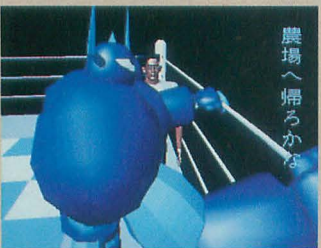
シャープX68000パソコン教室開催中
●会場：四谷教室
●コース：入門コース・表集計コース・音楽コース・絵画コース
●申込受付電話番号(03)3260-8365
●受講料：2,000円(税別)

夢、創ります。「第1回全日本X68000芸術祭」

クリエイティブマインドを刺激するビッグな規模のオリジナルソフトウェア・作品コンテストです。7月、四国より始まった全国10ヵ所の地区予選大会も大盛況のうちに終わり、平成4年4月に東京で開催される全国大会を待つばかりとなりました。ご期待ください。

食べ歩きもいいけど
グルメの本質は手料理だと思う。

特集　SX-WINDOWの未来

出たな!!　ツインビー

飛翔鮫

DōGA・CGA

吾輩はX68000である

サバイバル・ゲーム

# Oh!X

# C O N T



〈スタッフ〉
●編集長／前田 徹 ●副編集長／植木章夫 ●編集／岡崎栄子 浅井研二 山田純二 ●協力／有田隆也 中森 章 林 一樹 荻窪 圭 華門真人 毛内俊行 吉田賢司 影山裕昭 古村 聡 村田敏幸 丹 明彦 三沢和彦 長沢淳博 宮島 靖 金子俊一 浦川博之 石上達也 ●カメラ／杉山和美 ●イラスト／永沢しげる 山田晴久 寺尾響子 ●アートディレクター／島村勝頼 ●レイアウト／元木昌子 AD GREEN ●校正／グループごじら

表紙絵：塚田 哲也

# E N T S

1992 JAN.
1





■広告目次

**SHARP** システムパフォーマンスを実証する多彩なペリフェラル。

## ディスプレイ関連

### カラーディスプレイテレビ

14型カラーディスプレイテレビ
**CZ-607D-BK・-TN**
標準価格99,800円(税別)
(チルトスタンド同梱)

15型カラーディスプレイテレビ
**★CZ-605D-BK・-GY**
標準価格115,000円(税別)
(スピーカー2個・チルトスタンド同梱)

15型カラーディスプレイテレビ
**CZ-614D-BK・-TN**
標準価格135,000円(税別)
(スピーカー2個・チルトスタンド同梱)

### カラーディスプレイ

14型カラーディスプレイ
**CZ-606D-TN・-BK・-GY**
標準価格79,800円(税別)
(チルトスタンド同梱)

14型カラーディスプレイ
**CZ-604D-BK・-GY**
標準価格94,800円(税別)
(スピーカー2個・チルトスタンド同梱)

21型カラーディスプレイ
**CU-21HD**
標準価格148,000円(税別)
(スピーカー2個同梱)

### CRTフィルター

高性能CRTフィルター
**BF-68PRO**
標準価格19,800円(税別)
(14/15型用)

### チューナー

RGBシステムチューナー
**CZ-6TU-BK・-GY**
標準価格33,100円(税別)
(リモコン付)

## アートツール

### 画像入力

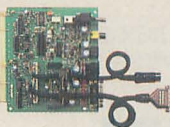
カラーイメージスキャナ※1
**CZ-8NS1**
標準価格188,000円(税別)

カラーイメージスキャナ※1
**JX-220X**
標準価格168,000円(税別)
※RS-232C/パラレルインターフェイス標準装備

スキャナ用パラレルボード
**CZ-6BN1**
標準価格29,800円(税別)

### 映像入力

カラーイメージユニット※2
**CZ-6VT1-BK**
**CZ-6VT1**
標準価格69,800円(税別)

### 映像出力

ビデオボード※3
**CZ-6BV1**
標準価格21,000円(税別)

## プリンタ

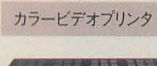
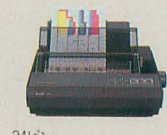
### 熱転写カラープリンタ

48ドット
熱転写カラー漢字プリンタ
**CZ-8PC5-BK**
標準価格96,800円(税別)
(信号ケーブル同梱)

### カラービデオプリンタ

カラービデオプリンタ
**★CZ-6PV1**
標準価格198,000円(税別)
(信号ケーブル同梱)

### カラーイメージジェット

カラーイメージジェット※4
**IO-735X-B**
標準価格248,000円(税別)
(信号ケーブル別売)
※グレータイプのIO-735Xもあります。

### カラードットプリンタ

24ピン
カラー漢字プリンタ(80桁)
**CZ-8PG1**
標準価格130,000円(税別)
(信号ケーブル同梱)

24ピン
カラー漢字プリンタ(136桁)
**CZ-8PG2**
標準価格160,000円(税別)
(信号ケーブル同梱)

### ドットプリンタ

24ピン漢字プリンタ(136桁)
**CZ-8PK10**
標準価格97,800円(税別)
(信号ケーブル同梱)

## ファイル

### 光磁気ディスク

光磁気ディスクユニット※5
(594MB)
**CZ-6MO1**
標準価格450,000円(税別)
(SCSIケーブル同梱)
※光磁気ディスクカートリッジは別売です。別売のJY-701MPA 標準価格30,000円(税別)をご使用ください。

### ハードディスク

増設用ハードディスクドライブ(40MB)
(CZ-602C/603C/652C/653C内蔵用)
**★CZ-64H**※
標準価格120,000円(税別)
(取付費別)

増設用ハードディスクドライブ(81MB)
(CZ-604C/634C内蔵用)
**CZ-68H**※
標準価格160,000円(税別)
(取付費別)
※取付に関してはシャープお客様ご相談窓口にてご相談ください。

ハードディスクユニット(20MB)
**★CZ-620H**
標準価格178,000円(税別)
※CZ-604C/623C/634C/644Cでは使用できません。

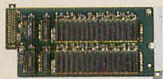
※1 ご使用に際しては、カラーイメージスキャナCZ-8NS1、JX-220Xに同梱のRS-232Cケーブルで接続するか、より高速のパラレルデータ伝送を行う場合、別売のスキャナ用パラレルボードCZ-6BN1標準価格29,800円(税別)で接続してください。※2 テレビチューナーを内蔵していないディスプレイをご使用の場合は、RGBシステムチューナーCZ-6TU(別売)が必要です。 ※3 ビデオ出力は15.75kHzテレビ標準信号です。また、拡張I/Oスロットは2スロット使用します。 ※4 別売の信号ケーブルIO-73CX 標準価格5,500円(税別)で接続してください。 ※5 CZ-600C、601C、602C、603C、611C、612C、613C、652C、653C、662C、663Cにご使用の場合は、別売のSCSIボード(CZ-6BS1)が必要です。また、X68000用 OS Human 68k ver 2.0以上にてご使用ください。(光磁気ディスクカートリッジは別売のJY-701MPA 標準価格30,000円(税別)をご使用ください。) ※6 ご使用に際しては、あらかじめ別売の1MB増設RAMボード CZ-6BE1 標準価格 35,000円(税別・

## お望みのパワーシステムへ。

## ボード

### 拡張メモリ

NEW
2MB増設RAMボード
(CZ-634C/644C専用)
**CZ-6BE2A**
標準価格59,800円(税別)
※2MB増設RAM(CZ-6BE2B)専用ソケットを2個用意しています。

NEW
2MB増設RAM
(CZ-634C/644C専用)
**CZ-6BE2B**
標準価格54,800円(税別)
※本増設RAM(CZ-6BE2B)は、2MB増設RAMボードが必要です。CZ-6BE2A上の専用ソケット(2個用意)に装着ください。
※取付に関してはシャープお客様ご相談窓口にてご相談ください。

1MB増設RAMボード
(CZ-600C専用)
★**CZ-6BE1**
標準価格35,000円(税別)

1MB増設RAMボード
(CZ-601C/611C/652C/653C/662C/663C用)
**CZ-6BE1B**
標準価格28,000円(税別)

2MB増設RAMボード※6
**CZ-6BE2**
標準価格79,800円(税別)

4MB増設RAMボード※6
★**CZ-6BE4**
標準価格138,000円(税別)

### インターフェイス

SCSIボード※7
**CZ-6BS1**
標準価格29,800円(税別)
(ソフトウェア(SCSIユーティリティ)同梱)

ユニバーサルI/Oボード
★**CZ-6BU1**
標準価格39,800円(税別)

GP-IBボード
★**CZ-6BG1**
標準価格59,800円(税別)

増設用RS-232Cボード
(2チャンネル)
★**CZ-6BF1**
標準価格49,800円(税別)

### MIDI

NEW
MIDIボード
**CZ-6BM1A**
標準価格26,800円(税別)

### FAX

FAXボード
**CZ-6BC1**
標準価格79,800円(税別)

### 数値演算プロセッサ

数値演算プロセッサボード
**CZ-6BP1**
標準価格79,800円(税別)

NEW
数値演算プロセッサ
(CZ-634C/644C専用)
**CZ-6BP2**
標準価格45,800円(税別)
※取付に関してはシャープお客様ご相談窓口にてご相談ください。
※特別ケース入りです。

## ネットワーク

### モデム

モデムユニット※8
**CZ-8TM2**
標準価格49,800円(税別)
(RS-232Cケーブル同梱)

### RS-232Cケーブル

RS-232Cケーブル
(平行接続型)
**CZ-8LM1**
標準価格7,200円(税別)

RS-232Cケーブル
(クロス接続型)
**CZ-8LM2**
標準価格7,200円(税別)

### LANボード

LANボード
**CZ-6BL1**
標準価格268,000円(税別)
(イーサネット用)

**CZ-6BL2**
標準価格298,000円(税別)
(イーサネット/チーパネット両用)
※電源ユニット・ソフトウェア
(ネットワークドライバVer1.0)同梱

## 入力

インテリジェントコントローラ
**CZ-8NJ2**
標準価格23,800円(税別)

マウス・トラックボール
**CZ-8NM3**
標準価格9,800円(税別)

トラックボール
**CZ-8NT1**
標準価格13,800円(税別)

マウス
**CZ-8NM2A**
標準価格6,800円(税別)

ジョイカード
**CZ-8NJ1**
標準価格1,700円(税別)

## その他

### 拡張スロット

拡張I/Oボックス(4スロット)
(CZ-600C/601C/602C/603C/604C/611C/612C/613C/623C/634C/644C用)
**CZ-6EB1-BK**
★**CZ-6EB1**
標準価格88,000円(税別)

### スピーカー

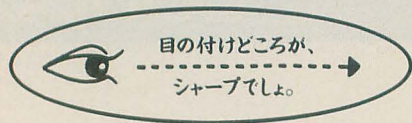
アンプ内蔵
スピーカーシステム(2本1組)
**AN-S100**
標準価格36,600円(税別)

### システムラック

システムラック
(CZ-600C/601C/602C/603C/604C/611C/612C/613C/623C/634C/644C用)
**CZ-6SD1**
標準価格44,800円(税別)

■本広告に掲載しております拡張ボード類のうち、CZ-634C/644Cの16MHzモードで動作しないものが一部あります。 ★印の商品は在庫僅少です。
■製品改良のため仕様の一部を予告なく変更することがあります。またこの広告の色調は印刷のため実物とは多少異なる場合もありますのであらかじめご了承ください。

CZ-600C用)、CZ-6BE1B 標準価格28,000円(税別・CZ-601C、CZ-611C、652C、653C、662C、663C用)を増設してください。 ※7 CZ-600C、601C、602C、603C、611C、612C、613Cに装着の場合、I/Oスロット2に装着ください。CZ-652C、653C、662C、663Cに装着の場合はI/Oスロット4に装着ください。また、CZ-6BG1、6BU1、6BL1、6BL2、6BN1などのボードは、接続コネクタとの関係で本ボードとの併用はできませんのでご注意ください。なお、本ボードはX68000用OS Human 68K ver.2.0以上にてご使用ください。 ※8 モデムユニットCZ-8TM2に同梱のソフトはX1/X1ターボシリーズ用です。

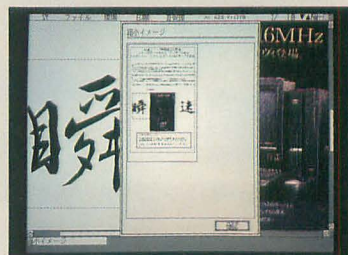
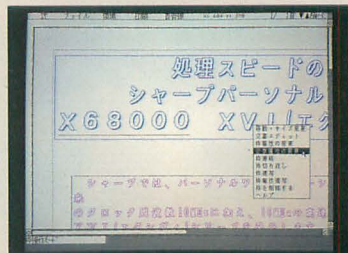
電子機器事業本部 AVCシステム事業推進室 〒162 東京都新宿区市谷八幡町8番地 ☎(03)3260-1161(大代表)

## 各種エディタを装備したレイアウトソフト

NEW

# PressConductor PRO-68K

CZ-266BS　標準価格28,000円(税別)　12月発売予定

簡単なマウス操作、まるで机の上で紙を貼り合わせる感覚で、文章、図形、罫線などをディスプレイ上で自由にレイアウトできます。㈱ツァイトの「書体倶楽部」全アウトラインフォント(明朝体、ゴシック体、毛筆体、教科書体)に対応。

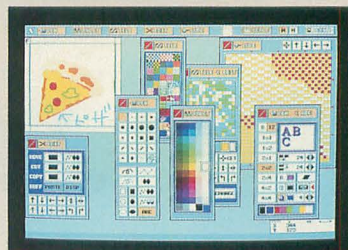
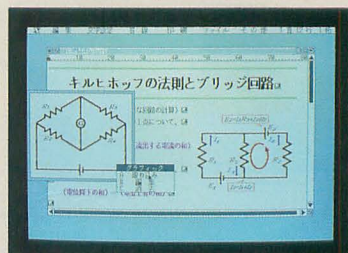
●袋文字、強調、回転、影文字、斜体字など豊富な文字装飾を装備●フォントの文字サイズ、文字装飾、文字色を1文字ごとに指定でき、緻密な文書編集が可能な「文書枠機能」●スキャナ読み込みや、Z'sSTAFF、New PirntShop PRO-68Kのデータファイルの読み込みも可能な「図形枠機能」●1024×1024ドットの大きな文字から24×24ドットの小さな文字まで、任意のサイズで文字が描ける「図形字枠機能」●カラー印刷対応●ページプリンタ対応
※メインメモリ2MB必要です。

## 多彩なグラフィック機能搭載多機能ワープロ

マルチワープロ PRO-68K

# Multiword

CZ-225BS　標準価格32,000円(税別)

WYSIWYGを採用したウィンドウモード、エディタ感覚で入力できるテキストモード。さらにクリエイティブマインドを刺激する多彩なグラフィック機能を搭載。X68000のパフォーマンスをフルに活かしたヒューマンワープロの誕生です。

●最大10文書までの複数文書を同一画面で編集可能。文書間のカット&ペーストも可能●スピーディな文字入力をサポートするテキストモード●20種類のペンを装備したグラフィックエディタを装備●多彩な文字種、文字間隔もドット単位に指定可能●豊富な改行、罫線機能●用途に合わせて選べる幅広いプリンタサポート、多彩な用紙設定、印字設定が可能●イメージスキャナ入力はパラレルインターフェイスに対応。ハンディスキャナ入力もサポート。
※メインメモリ2MB必要です。

## 「Multiword」発売記念キャンペーン実施中!!

「Multiword」CZ-225BSの発売を記念し、期間中にご購入の方にステキな賞品が当ります。この機会にぜひご購入ください。

- 期間：平成3年8.1～12.31迄(消印有効)
- 対象：「Multiword」ご購入ユーザーで登録カードを弊社に送付された方の中から、厳正な抽選により決定いたします。
- 発表：パソコン専門誌X68000ソフト広告誌上で発表します。
- 賞品：1等/X68000フロッピーアタッシュケース……5名
  2等/X68000シースルークロック……10名
  3等/X68000キーホルダー/ネクタイピン……15名
  4等/X68000マイウェイバッグ……20名
  5等/X68000特製テレフォンカード/ステッカー……30名

# X68000 APPLICATION REVIEW

## MONTHLY PICK UP

●シューティングゲーム

### 中華大仙

CZ-268AS　標準価格7,900円(税別)

©TAITO CORP. 1988

●コミカルアクションゲーム

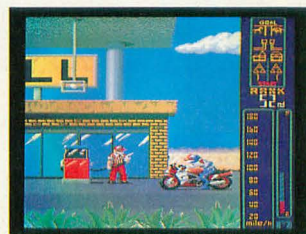
### ボナンザブラザーズ

CZ-270AS　標準価格9,000円(税別)

©SEGA1990 REPROGRAMMED BY SHARP/SPS
※メインメモリ2MB必要です。

●バイクレーシングゲーム

### ダッシュ野郎

CZ-269AS　標準価格8,800円(税別)

©TOAPLAN Co. Ltd. 1988

●高速カード型リレーショナルデータベース

CZ-253BS 標準価格29,800円(税別)

操作性の向上、高速化を図った新マルチウィンドウシステムを搭載したニューバージョンです。一覧表画面入力、グラフ機能などをサポート。キーボード操作にも対応します。

※メインメモリ2MB必要です。
※CARD PRO-68K(CZ-226BS)をお持ちの方には有償バージョンアップを行います。

**CARD PRO-68K ver2.0用 パーソナルプログラム集** NEW
CZ-276BS　標準価格12,000円(税別)

**CARD PRO-68K ver2.0用 ビジネスプログラム集** NEW
CZ-279BS　標準価格12,000円(税別)

●ビジネスグラフチャートソフト

CZ-267BS　2月発売予定

各種データベースで作成したデータをもとに、多彩なグラフが作成できます。3次元表示やグラフの複合機能も装備。また作成したグラフを文章とレイアウトすることもでき、プレゼンテーションや経営シミュレーションなどに活用できます。

●Zeit日本語ベクトルフォントをサポート

CZ-265HS 標準価格20,000円(税別)

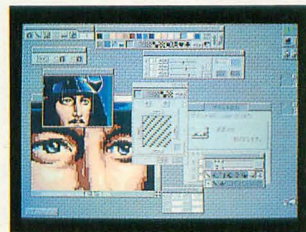
効率のよい操作環境を実現。カセットレーベル、カレンダー作成に対応したほか、モノクロデータの編集などグラフィックエディタを強化した高機能テキストエディタを内蔵しています。

※メインメモリ2MB必要です。
※NEW Print Shop PRO-68K(CZ-221HS)をお持ちの方には有償バージョンアップを行います。

**グラフィックライブラリ VOL.3** NEW
CZ-283GS　標準価格8,000円(税別)

●SX-WINDOW対応ペイントツール

### Easypaint SX-68K

CZ-263GW 標準価格12,800円(税別)

マウスによる簡単操作、65,536色中16色の多彩なカラー表現、SX-WINDOW対応初のペイントツールです。同時に複数のウィンドウを開いて編集でき、各ウィンドウ間でデータのやりとりもOK。

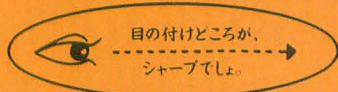
※メインメモリ2MBおよびSX-WINDOW ver.1.1が必要です。

**SX-WINDOWイラスト集 VOL.1** NEW
CZ-280GW　標準価格8,000円(税別)

**SX-WINDOWイラスト集 VOL.2** NEW
CZ-281GW　標準価格8,000円(税別)

《お詫びと訂正》Oh!X 12月号付録の「Press Conductor PRO-68K」のソフト広告において、一部対応プリンタ・スキャナ機種に誤りがありますので訂正させていただくとともに、謹んでお詫び申し上げます。
◆〈対応プリンタ〉……(誤)SHARP CZ-8PX5→(正)SHARP CZ-8PK5
◆〈対応スキャナ〉……(誤)SHARP CZ-BNS1→(正)SHARP CZ-8NS1

※CZ-253BS、CZ-265HSへの有償バージョンアップについては、下記にお問い合わせください。

●お問い合わせは…シャープ㈱電子機器事業本部AVCシステム事業推進室 〒162 東京都新宿区市谷八幡町8番地 ☎(03)3260-1161(大代表)へ。 シャープ株式会社

「夢、創ります。山下章氏プロデュース」

# 第1回全日本X68000 芸術祭

■主催：シャープ株式会社 電子機器事業本部 システム機器営業部
■共催：シャープエレクトロニクス販売株式会社各統轄営業部
東京中央シャープ販売(株)・浪速シャープ電機(株)・沖縄シャープ電機(株)
■協賛：出版社・ソフトハウス・サードパーティ・主要販売店

全国のユーザーを熱くさせているX68000芸術祭地区大会も、大盛況のうちに終了いたしました。多数の作品ご応募ありがとうございました。ここに紹介される大賞作品・入選作品は、平成4年4月に東京で開催される全国大会で、グランプリを競い合います。どの作品が日本一の栄冠に輝くか、まだまだ目が離せません。

LOGICRUSH（四国）　TVinTV（中国）　白セン菌くん（中国）　TORNADO（神奈川）　CYNTHIA（東北）　SX-MEGATONE（北海道）　C力検査（北関東）　RUSH！（近畿）　USEFUL（近畿）

FORTRESS ATTACK（北陸）

EYE（近畿）

《ミュージック部門受賞者》

見上げてごらん（中部）

Turn on the Run（中部）

| 開催地区 | 賞 | 受賞作品名 | 部門 | 機種 | 受賞者名（敬称略） |
|---|---|---|---|---|---|
| 四国地区大会 | 大賞 | LOGICRUSH | ゲーム | X68000 | 鴨居大吾・藤原智行 |
| 北海道地区大会 | 大賞 | SX-MEGATONE | その他 | X68000 | 濱田淳一 |
| 東北地区大会 | 大賞 | CYNTHIA | ゲーム | X68000 | 今橋晃一 |
| 中国地区大会 | 大賞 | TV in TV | グラフィック | X68000 | 前田　浩 |
| 〃 | 特別入選 | 白セン菌くん | その他 | X68000 | 藤本幹雄 |
| 北関東地区大会 | 大賞 | C力検査 | ゲーム | X68000 | 小林康弘 |
| 神奈川地区大会 | 大賞 | TORNADO | グラフィック | X68000 | 文月　涼 |
| 中部地区大会 | 大賞 | 見上げてごらん | ミュージック | X68000＋CM-64 | 寺田光太郎 |
| 〃 | 入選 | Turn on the Run | ミュージック | X68000＋etc. | 伊藤忠彦 |
| 北陸地区大会 | 大賞 | FORTRESS ATTACK | ゲーム | X68000 | 柴原章宏 |

| 開催地区 | 賞 | 受賞作品名 | 部門 | 機種 | 受賞者名（敬称略） |
|---|---|---|---|---|---|
| 近畿地区大会 | 大賞 | RUSH！ | ゲーム | X68000＋PC-9801 | 京大マイコンクラブ |
| 〃 | 入選 | EYE | グラフィック | X68000 | 鎌田優・古本隆行 |
| 〃 | 入選 | USEFUL | その他 | X68000 | 荒田隆仁 |
| 首都圏地区大会 | 大賞 | PENGUIN ランドネット OFF会記念 | ゲーム | X68000 | 竹内久徳 |
| 〃 | 入選 | レイトレアニメ4 | グラフィック | X68000 | 木村哲也 |
| 〃 | 入選 | BALANCER PRO68K | ゲーム | X68000 | 伊藤英介 |
| 〃 | 入選 | ダンスディスク | その他 | PC-8801 | 唐沢幸一・金井毅 |
| 九州地区大会 | 大賞 | ※ | | | |
| 〃 | 入選 | ※ | | | |

※九州地区大会の受賞作は、次号をご覧ください。

**補選のお知らせ LAST CHANCE**

各地区大会応募締め切りに間に合わなかった方のために、補選を行います。
作品審査をいたしますので、個性あふれる作品をドシドシお送りください。応募締め切りは、平成4年1月24日㈮必着です。
（応募・問い合わせ）〒545 大阪市阿倍野区長池町22-22 シャープ株式会社電子機器事業本部システム機器営業部「X68000芸術祭」係 ☎06-621-1221㈹

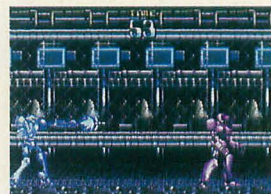
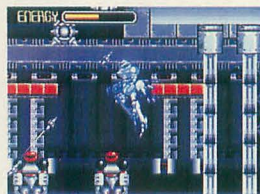
●お問い合わせは…シャープ㈱電子機器事業本部システム機器営業部 〒545 大阪市阿倍野区長池町22番22号 ☎(06)621-1221（大代表） シャープ株式会社

zeit
グラコンが、はじまったよ。
いつの日か、君がふと描いていたグラフィックス。
いまでもデータの中に残っているんじゃないかな。
いよいよ公開するときがきたね。
ツァイトが開催する第1回ジーズ・グラフィックス・コンテスト。
あなたのオリジナル作品を、求めます。
ウマイ。ヘタ。なんてくだらない一辺倒な審査はしない。
これはどちらかというと、フェスティバルだ。
ウンと楽しいお祭り。参加しない手は、ないと思うよ。
ミス・グラコン
Z's GraCon
■応募要領
応募には次の4つの部門を用意。
◎年賀状に使ったお気に入りの絵で参加できる――年賀状部門
◎彼女や彼氏、有名人の似顔絵で参加できる――似顔絵部門
◎バイクや戦闘機などの乗り物の絵で参加できる――乗り物部門
◎元気のいいグラフィックスで参加できる――自由課題部門
みんなの遠慮のないエントリーをお待ちしています。また入賞者および参加者全員には素敵なプレゼントをご用意。詳しくは有名パソコンショップもしくはZ'sGRACON事務局までお問い合わせください。資料を至急お送りします。
■お問い合わせ・作品の送り先
株式会社ツァイト
第1回Z'sGRACON事務局
〒151東京都渋谷区初台1-47-1
小田急西新宿ビル4F
TEL03-3299-0461
※第1回ジーズ・グラフィックス・コンテストは、SHARPのX68000にのるジーズスタッフPRO-68K/「Ver.2.0」とジーストリフォニィ「デジタルクラフト」による作品のみに限らせていただきます。
ツァイトが開催する第1回ジーズ・グラフィックス・コンテスト。
あなたの作品募集中
[Xのためのグラコン・パック] SHARPのX68000にのるジーズスタッフPRO-68K/「Ver.2.0」とジーストリフォニィ「デジタルクラフト」、書体倶楽部「毛筆体JIS第1水準」、さらにはサンプルディスク(イメージや3Dデータ)が1パックになって、特別記念価格でなんと¥68,000(税別)。1992年1月31まで、好評発売中!あなたもこの期間にぜひ。応募も、忘れずにねッ。

brother

## 熱い興奮!! 君は宇宙最強戦士『Heavy Nova』(ヘビーノヴァ)になれるか!?

地球圏防衛軍の中核を為す部隊『Heavy Dool』隊。
そのパイロット養成所から今、壮大な物語が始まろうとしている
目的はただ1つ!宇宙で偉大な戦士だけに与えられる称号
『Heavy Nova』を獲得することだ!君はHeavy Doolを操り、
果たして宇宙最強の戦士になれるか?

友達とも対戦できる!2人プレイ可能!!

## 2月10日発売予定!

価格 ¥5,800(税込) ■対応機種：X68000 ■企画/開発：㈱マイクロネット

「邪神ダベダの復活は誰にも邪魔させん……」

軽快な戦闘、高い操作性

スピーディなストーリー展

X68000版 好評発売中

価格¥5,900(税込)

■対応機種：X68000シリーズ(3枚組)
■企画/開発：アルファ・システム

メガドライブで大人気のピンボールゲーム「DINO LAND(ディノランド)」をX68000に移植。手軽に遊べて、かつおもしろいゲームの誕生です。「羽田一郎」プロデュースのBGMにのせて跳ねまわるディノくんにぜひ会ってみて、ネ!

価 格¥7,800(税込)

■対応機種：X68000 ■企画/開発：ウルフチーム



ブラザー工業株式会社 〒467 名古屋市瑞穂区苗代町2番1号
TAKERU事務局 (052)824-2493
東京営業所(03)3274-6916
大阪営業所(06)252-4234

通信販売 通信販売をご希望の方は、ソフト名・機種名・住所・氏名・電話番号を明記の上TAKERU事務局まで現金書留でお申し込み下さい
代金引換は一度、現金書留で申し込んで頂いた方に案内させて頂きます

スーパーマルチキャラクターA.R.P.G.
ファーストクイーン

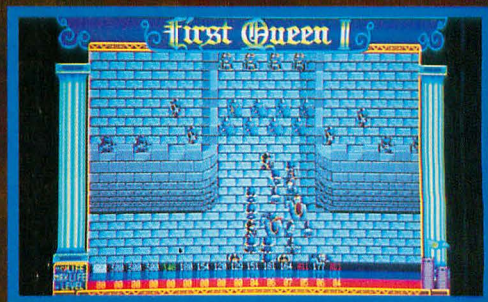
# First Queen II

——砂漠の女王——

illustration：天野喜孝

おまたせしました X68000 完全移植版

12月14日発売予定 8,800円

＊都合により発売が延期になりましたことをお詫び申し上げます。

ヒーローをさがせ!!

**K.S.K.ゴチャキャラシリーズ**

| | | |
|---|---|---|
| R.P.G. シルバーゴースト（元祖ゴチャキャラ） | PC-88X1シリーズ | 8,800円 |
| R.P.G. ファーストクイーン | PC-98X68000シリーズ | 8,800円 |
| R.P.G. ファーストクイーンII | PC-98/286シリーズ | 8,800円 |
| S.L.G. DUEL | PC-88シリーズ | 8,700円 |
| S.L.G. DUEL MC対応版 | PC-88シリーズ | 9,000円 |
| S.L.G. DUEL川中島シナリオ（セミオート付） | タケル専用 | 3,800円 |
| S.L.G. DUEL 98（2部隊コントロール可能） | PC-98シリーズ | 8,700円 |

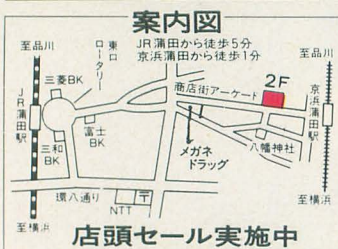
お求めはお近くのパソコンショップで。通信販売をご希望の方は住所・氏名・電話番号・商品名・機種名を明記の上、現金書留でお申し込み下さい。

■店頭にて、新作ゲームソフト25～30%OFF!!（税別）、超低金利オクトハッピークレジットをご利用下さい!!

■翌月末一括（1月末）払いOK!!手数料無料!!ご利用下さい。■店頭にて、新作ゲームソフト25～30%OFF!!

■便利です。夜9時まで営業しております。お立寄り下さい。お待ちしております!!

パソコンプラザ

案内図

店頭セール実施中

## オクトで始まるパソコンワールド

☎03-3730-6271

●営業時間 AM 11:00～9:00/日曜・祭日PM7:00　電話一本で、ハイ即納

〒144 東京都大田区蒲田4-6-7 FAX 03-3730-6273

全国通販 ●定休日毎週火曜日 祭日の場合翌日になります。

オクト ラクラククレジット

| 3回 | 3.5 | 6回 | 4.5 | 10回 | 6.0 | 12回 | 6.0 | 15回 | 9.0 | 18回 | 11.0 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 20回 | 12.0 | 24回 | 12.5 | 30回 | 17.0 | 36回 | 17.5 | 48回 | 23.0 | 60回 | 33.0 |

OCT-1 システム インフォメーション

▶全商品保証付（メーカー保証）
▶超低金利ハッピークレジット（1回～60回）頭金ナシOK！
▶ボーナス一括払いOK！ボーナス2回払いOK!!
▶配達日の指定OK！（万全なサポート体制）
▶商品の組合せ自由！ オクトフリーダムシステム
▶店頭デモンストレーション実施中

オクト セレクテッドシステム

広告掲載商品以外の製品も取扱っております。

蒲田

## 1991年末～1992年始★限定販売★
## ―今が買い頃！冬期限定大セール中!!―

SHARP

X68000 XVI エクシヴィ

■16MHz■
■SX-WINDOW ver1.1■
■Attachment MEMORY BORD■

快速16MHz 鮮烈デビュー!!

### ■CZ-634C-TN（定価￥368,000）

Ⓐ●CZ-634C-TN
●CZ-614D-TN NEW
定価合計￥503,000 ▶特価TEL下さい。

| 12回 | ￥32,700 | 24回 | ￥17,200 | 36回 | ￥11,900 | 48回 | ￥9,300 |
|---|---|---|---|---|---|---|---|

Ⓑ●CZ-634C-TN
●CZ-607D-TN NEW
定価合計￥467,800 ▶特価TEL下さい。

| 12回 | ￥30,700 | 24回 | ￥16,300 | 36回 | ￥11,300 | 48回 | ￥8,900 |
|---|---|---|---|---|---|---|---|

Ⓒ●CZ-634C-TN
●CZ-606D-TN
定価合計￥447,800 ▶特価TEL下さい。

| 12回 | ￥29,100 | 24回 | ￥15,400 | 36回 | ￥10,600 | 48回 | ￥8,300 |
|---|---|---|---|---|---|---|---|

### ■CZ-644C-TN（定価￥518,000）

Ⓓ●CZ-644C-TN
●CZ-614D-TN NEW
定価合計￥653,000 ▶特価TEL下さい。

| 12回 | ￥42,300 | 24回 | ￥22,300 | 36回 | ￥15,500 | 48回 | ￥12,100 |
|---|---|---|---|---|---|---|---|

Ⓔ●CZ-644C-TN
●CZ-607D-TN NEW
定価合計￥617,800 ▶特価TEL下さい。

| 12回 | ￥40,400 | 24回 | ￥21,400 | 36回 | ￥14,900 | 48回 | ￥11,700 |
|---|---|---|---|---|---|---|---|

Ⓕ●CZ-644C-TN
●CZ-606D-TN
定価合計￥597,800 ▶特価TEL下さい。

| 12回 | ￥38,700 | 24回 | ￥20,500 | 36回 | ￥14,200 | 48回 | ￥11,100 |
|---|---|---|---|---|---|---|---|

### X68000XVI ドッカ～ン！プレゼント!!

―あなたのオクトから素敵な贈物―

今、XVIをお買い上げいただいた方は、プレゼントの①番か②番のどちらかお選び下さい。プラス③番はもれなくプレゼント!!

① 大人気

生中継68 野球ゲームの決定版

大戦略II（キャンペーン版） 不朽の名作X68000版

新作！大人気ゲームソフト!!

（定価￥9,800）　（定価￥9,800）

or

② インテリジェントコントローラ
■CZ-8NJ2（CYBER STICK）
シューティングゲーマーの必須アイテム!!

（定価￥23,800）

③ MD-2HD（10枚）
シリコンキーボードカバー
もれなく!! サービス!!

▶現金超特価 ￥TEL下さい!!▶

※どちらかお選び下さい!!（どっちが得かヨーク考えてネ！）

### 特選周辺機器（送料￥500）

●SX-68M MIDIインターフェースボード（システムサコム）￥19,800…特価￥13,600
●Fine Scanner X68（HAL研究所）（HGS-68）￥39,800…特価￥25,200

■増設RAMボード＝I・Oデータ

① PIO-6BE1-A（1MB） ￥25,000…特価￥16,000
② PIO-6BE2-2M（2MB） ￥50,000…特価￥31,800
③ PIO-6BE4-4M（4MB） ￥88,000…特価￥55,000

### 周辺機器コーナー（送料無料）

●CZ-6BEI IBM増設RAMボード（￥35,000）▶特価￥26,250
●CZ-6BEIB IBM増設RAMボード（￥28,000）▶特価￥21,000
●CZ-6BE2 2MB増設RAMボード（￥79,800）▶特価￥59,850
●CZ-6BE4 4MB増設RAMボード（￥138,000）▶特価￥103,500
●CZ-6BFI 増設用RS-232Cボード（￥49,800）▶特価￥37,350
●CZ-6BGI GP-IBボード（￥59,800）▶特価￥44,850
●CZ-6BMI MDIボード（￥26,800）▶特価￥20,100
●CZ-6BNI スキャナ用パラレルボード（￥29,800）▶特価￥22,350
●CZ-6BPI 数値演算プロセッサボード（￥79,800）▶特価￥59,850
●CZ-6BOI ユニバーサルI/Oボード（￥39,800）▶特価￥29,850
●CZ-6EBI/BK 拡張I/Oボックス（￥88,000）▶特価￥66,000
●CZ-6VTI/BK カラーイメージ・ユニット（￥69,800）▶特価￥52,350
●CZ-8NM2A マウス（￥6,800）▶特価￥5,100
●CZ-8NTI マウストラックボール（￥9,800）▶特価￥7,350
●CZ-8NSI カラーイメージスキャナ（￥188,000）▶特価￥140,000
●CZ-6BCI FAXボード（￥79,800）▶特価￥59,850
●CZ-8TM2 モデムユニット（￥49,800）▶特価￥37,350
●CZ-64H 増設ハードディスク（￥120,000）▶特価￥90,000
●CZ-6TU GY/BK RGBシステムチューナー（￥33,100）▶特価￥24,800
●BF-68PRO 高性能CRTフィルター（￥19,800）▶特価￥14,850
●CZ-6MOI 光磁気ディスクユニット（￥450,000）▶特価￥337,500
●CZ-6BSI SCSIインターフェースボード（￥29,800）▶特価￥22,350
●CZ-6BL2 LANボード（￥298,800）▶特価￥223,500
●CZ-6BVI（ビデオボード）（￥21,000）▶特価￥15,750
●CZ-6BE2A 2MB増設RAMボード（￥59,800）▶特価￥44,850
●CZ-6BE2B 2MB増設メモリ（チップ型）（￥54,800）▶特価￥41,100
●CZ-6BP2 数値演算プロセッサ（￥45,800）▶特価￥34,350
●AN-S100 スピーカーシステム（2本1組）（￥36,600）▶特価￥27,450

※クレジットの回数は1回～60回、ボーナス併用などありますのでお電話でお問合せ下さい。

■本体セット：送料無料（注）本体セット以外の周辺機器（プリンター、モデム、HDD等）及びソフトの送料は、北海道・九州地区＝1ケロ￥1500、■その他離島地区は、1ケロ￥2000となります。
※上記料金には、消費税は含まれておりません。消費税が付加されますので、詳しくは、電話でお問合せ下さい。

■特に人気のある商品によっては、しばらくお待ち願うことがありますのでご了承下さい!!

# X68000 ラストチャンス!!

## ■SUPER/PROII/SUPER-HD

生中継68
野球ゲームの決定版

(定価￥9,800)

プレゼント

新作!大人気ゲームソフト!!

大戦略II
(キャンペーン版)
不朽の名作X68000版

(定価￥9,800)

さらに! ★JOY CARD(連射式)×2個
さらにさらに!! ★MD-2HD 10枚

■超低金利クレジットをご利用下さい。1回~60回払い、頭金ナシ!!ボーナス1回及び2回払いOKです。

■ビッグバーゲンセール実施中!!ゲームソフト(ビジネス)新製品続々入荷中!!

限定

■SUPER(定価￥348,000)
CZ-604C-TN

■PRO II(定価￥285,000)
CZ-653C-BK/GY

■SUPER-HD(定価￥498,000)
CZ-623C-TN

15型カラーディスプレイTV

CZ-614D-TN
定価￥135,000

14型カラーディスプレー

CZ-606D(GY/BK/TN)
定価￥79,800

21型カラーディスプレイ

CU-21HD
定価￥148,000

CZ-8NJ2 限定
●インテリジェントコントローラ
定価￥23,800
超特価¥18,000

Ⓐ CZ-604C+CZ-614D……定価合計￥483,000▶ ¥306,000

| 12回 | ￥27,800 | 24回 | ￥14,700 | 36回 | ￥10,200 | 48回 | ￥8,000 | 60回 | ￥6,900 |
|---|---|---|---|---|---|---|---|---|---|

Ⓑ CZ-653C+CZ-614D……定価合計￥420,000▶ ¥279,000

| 12回 | ￥25,300 | 24回 | ￥13,400 | 36回 | ￥9,300 | 48回 | ￥7,300 | 60回 | ￥6,300 |
|---|---|---|---|---|---|---|---|---|---|

Ⓒ CZ-623C+CZ-614D 定価合計￥633,000▶ ¥366,000

| 12回 | ￥33,200 | 24回 | ￥17,600 | 36回 | ￥12,300 | 48回 | ￥9,600 | 60回 | ￥8,300 |
|---|---|---|---|---|---|---|---|---|---|

Ⓓ CZ-604C+CZ-606D……定価合計￥427,800▶ ¥268,000

| 12回 | ￥24,300 | 24回 | ￥12,900 | 36回 | ￥9,000 | 48回 | ￥7,000 | 60回 | ￥6,100 |
|---|---|---|---|---|---|---|---|---|---|

Ⓔ CZ-653C+CZ-606D……定価合計￥364,800▶ ¥218,000

| 12回 | ￥19,800 | 24回 | ￥10,500 | 36回 | ￥7,300 | 48回 | ￥5,700 | 60回 | ￥4,900 |
|---|---|---|---|---|---|---|---|---|---|

Ⓕ CZ-623C+CZ-606D……定価合計￥577,800▶ ¥328,000

| 12回 | ￥28,900 | 24回 | ￥15,300 | 36回 | ￥10,700 | 48回 | ￥8,400 | 60回 | ￥7,200 |
|---|---|---|---|---|---|---|---|---|---|

Ⓖ CZ-604C+CU-21HD 定価合計￥496,000▶ ¥313,000

| 12回 | ￥28,400 | 24回 | ￥15,100 | 36回 | ￥10,500 | 48回 | ￥8,200 | 60回 | ￥7,100 |
|---|---|---|---|---|---|---|---|---|---|

Ⓗ CZ-653C+CU-21HD 定価合計￥433,000▶ ¥263,000

| 12回 | ￥23,900 | 24回 | ￥12,600 | 36回 | ￥8,800 | 48回 | ￥6,900 | 60回 | ￥6,000 |
|---|---|---|---|---|---|---|---|---|---|

Ⓘ CZ-623C+CU-21HD 定価合計￥646,000▶ ¥373,000

| 12回 | ￥33,900 | 24回 | ￥18,000 | 36回 | ￥12,500 | 48回 | ￥9,800 | 60回 | ￥8,500 |
|---|---|---|---|---|---|---|---|---|---|

(送料無料・税別)★クレジット価格は、消費税込みですヨ〜!ご利用下さい!!

## X68000ソフト大セール実施中!!(ゲームソフト25~30%OFF) (送料￥500)

〈グラフィック〉●Z's STAFF PRO68K Ver.2.0
(シャフト)定価￥58,000
…………… 特価¥37,800

〈開発ツール〉●C-コンパラPRO68KV.2
定価￥44,800 CZ-245IS
…………… 特価¥32,800

〈データベース〉●CARD PRO68K Ver.2.0
定価￥29,800 CZ-253BS
…………… 特価¥20,800

〈グラフィック〉●C-TRACE 68 Ver.3.0
定価￥98,000
…………… 特価¥69,000

〈C言語〉●C & Professional Pack
定価￥58,000
…………… 特価¥40,000

〈音楽〉●Music studio PRO68K Ver. 2.0
定価￥28,800 CZ-261MS
…………… 特価¥21,200

〈CGシール〉●CANVAS PRO68K
定価￥29,800 CZ-249GS
…………… 特価¥22,200

〈ワープロ〉●Multiword PRO68K
定価￥32,000 CZ-225BS
…………… 特価¥23,800

〈通信〉●Tlepotion PRO68K
定価￥22,800 CZ-258BS
…………… 特価¥17,000

| 型名 | 商品 | 定価 | 特価 |
|---|---|---|---|
| CZ-212BS | BUSINESS PRO-68K | ￥68,000 | ￥48,000 |
| CZ-213MS | MUSIC PRO68K | ￥18,800 | ￥13,400 |
| CZ-214MS | SOUND PRO-68K | ￥15,800 | ￥11,400 |
| CZ-215MS | Sampling PRO-68K | ￥17,800 | ￥12,800 |
| CZ-219SS | OS-9/X68000 | ￥29,800 | ￥21,000 |
| CZ-220BS | DATA PRO-68K | ￥58,000 | ￥41,000 |
| CZ-223CS | Communication PRO-68K | ￥19,800 | ￥14,200 |
| CZ-224LS | THE 福袋 V2.0 | ￥9,900 | ￥7,500 |
| CZ-241BS | システム手帳リフィル集 | ￥9,800 | ￥7,500 |
| CZ-242BS | 活用フォーム集 | ￥9,800 | ￥7,500 |
| CZ-244SS | Homan 68K Ver2.0 | ￥9,800 | ￥7,500 |
| CZ-247MS | MUSIC PRO-68K(MIDI) | ￥28,800 | ￥20,800 |
| CZ-240BS | Stationery PRO-68K | ￥14,800 | ￥11,500 |
| CZ-243BS | CYBER NOTE PRO-68K | ￥19,800 | ￥15,200 |

| 型名 | 商品 | 定価 | 特価 |
|---|---|---|---|
| | Z's TRIPHNY(デジタルクラフト) | ￥39,800 | ￥27,300 |
| | テラッツオ(ハミングバード) | ￥19,400 | ￥13,800 |
| | KAMIKAZE(サムシンググッド) | ￥68,000 | ￥44,500 |
| | Final Ver.3.2(エーエスピー) | ￥38,000 | ￥29,500 |
| | サイクロンEXPRESSα68 | ￥98,000 | ￥69,500 |
| | G'ツール(ザインソフト) | ￥28,000 | ￥18,800 |
| | たーみのる2(SPS) | ￥17,800 | ￥13,200 |
| | G68K Ver.2 PRO | ￥22,000 | ￥17,500 |
| CZ-259SS | SX-WINDOW Ver.1.0 | ￥6,800 | ￥5,000 |
| CZ-251BS | ハイパーワード | ￥39,800 | ￥29,600 |
| CZ-260LS | XBAS to CHECKER PRO68K | ￥9,800 | ￥7,500 |
| CZ-234LS | AI-68K | ￥188,000 | ￥139,000 |
| CZ-255GS | CANVASドローグラフィックLiB | ￥8,800 | ￥6,600 |
| CZ-256GS | CANVASドローグラフィックVol.2 | ￥8,800 | ￥6,600 |

## ハイパー・ハードディスク (送料￥1,000)

■システムサコム
Mocking Bird SCSI
X68000/TOWNS用

■HD-J040(42M/25ms)
(￥89,000)▶大特価￥68,000
■HD-J100(100M/20ms)
(￥128,000)▶大特価￥94,000
■HD-J130(130M/20ms)
(￥148,000)▶大特価￥112,000
■HD-J170(173M/20ms)
(￥189,000)▶大特価￥128,000
※別売(SCSIカード)
FMT-121(￥30,000)特価¥21,500

## ハードディスク (送料￥1,000)

■アイテック
X68000用ハードディスク

● TX-80(定価￥108,000)….▶大特価￥77,000
(80MB、SCSI、SASI両対応)
● TX-130(定価￥138,000)…▶大特価￥95,000
(130MB、SCSI対応)
● TX-180(定価￥185,000)…▶大特価￥129,000
(180MB、SCSI対応)
※CZ-6BSI(SCSIボード)￥29,800‥▶特価¥22,350

## パソコンラック〈送料無料〉

Ⓐ5段キャスター付
スライド式キーボード台
● 1150(H)×640(W)×600(D)
定価￥38,000
特価 ¥12,500

Ⓑ4段キャスター付
● 1250(H)×640(W)×700(D)
定価￥29,800
特価 ¥8,800

## 店頭新作ゲームソフト25~30%OFF!! ビジネスソフト25%より特価中

★通信販売お申込みのご案内★ 〒144 東京都大田区蒲田4-6-7 TEL:03-3730-6271

お申込みはお電話でお願いします。お客様の〈住所〉〈氏名〉〈電話番号〉及び〈商品名〉をお知らせ下さい。● 入金確認後ただちに商品をご送付いたします。

**現金一括払い**
銀行振込:お近くの銀行より(電信扱い)にてお振込み下さい。
現金書留:封筒の中に住所・氏名・商品名をご記入の上当社までお送り下さい。

**クレジット**
専用お申込用紙をお送り致しますので、必要事項をご記入、ご捺印の上ご返送下さい。手続きは簡単です。

オクト ラクラク クレジット表

| 回数 | 料率 | 回数 | 料率 | 回数 | 料率 | 回数 | 料率 |
|---|---|---|---|---|---|---|---|
| 3回 | 3.5 | 6回 | 4.5 | 10回 | 6.0 | 12回 | 6.0 |
| 15回 | 9.0 | 18回 | 11.0 | 20回 | 12.0 | 24回 | 12.5 |
| 30回 | 17.0 | 36回 | 17.5 | 48回 | 23.0 | 60回 | 33.0 |

**振込先**
富士銀行 久ヶ原(クガハラ)支店 当No.1824
三菱銀行 蒲田支店 当No.0278691
株式会社 億人(オクト)

※掲載の価格は変動しますので、まずは、お電話にてご確認ください。
※上記料金には、消費税は含まれておりません。消費税が付加されますので、詳しくは電話でお問合せ下さい。
※銀行振込、または、現金書留でご注文の際には、あらかじめ電話でご確認の上、お申し込み下さい。

超低金利クレジットOK!

# 注目!!

平成4年3月末一括払い
手数料(金利)無料

(平成3年12月末はもちろんのこと
平成4年1月末/2月末/3月末のいずれかをご指定下さい。)

# またまた秋葉原でおなじみの

## 12/15~1/15

●お近くの方はお
●本体単品で特
●ビジネスソフト定

《増設メモリー&数値演算プロセッサ》計測技研

1 PRKII-02(2M)……定価￥ 55,000▶特価￥ 41,900
2 PRKII-04(4M)……定価￥ 90,000▶特価￥ 69,000
3 PRKII-06(6M)……定価￥125,000▶特価￥ 95,500
4 PRKII-08(8M)……定価￥160,000▶特価￥122,000
5 PRKII-12(2M)……定価￥ 85,000▶特価￥ 65,500
6 PRKII-14(4M)……定価￥120,000▶特価￥ 92,000
7 PRKII-16(6M)……定価￥155,000▶特価￥118,000
8 PRKII-18(8M)……定価￥190,000▶特価￥145,000
9 MC-68881RC……定価￥ 38,000▶特価￥ 28,000

カラーイメージジェット
■IO-735X-B
定価￥248,000
特価￥159,000
(送料・消費税込み￥164,800)

カラーイメージスキャナ
■JX-100S
定価￥89,800
特価￥44,000
(送料・消費税込み￥46,350)

■SX-68MII (MIDI)
(サコム)定価￥19,800
特価￥13,500
(送料・消費税込み￥14,420)

■HGS-68(スキャナ)
(HAL研)定価￥39,800
特価￥25,000
(送料・消費税込み￥26,265)

X68000メモリボード(I/O・DATA) (送料￥500)

① SH-6BE1-1M(600CE用) 定価￥25,000
(送料・消費税込み￥19,364)……特価￥18,300
② PIO-6BE1-A 定価￥25,000
(送料・消費税込み￥16,789)……特価￥15,800
③ PIO-6BE2-2M 定価￥50,000
(送料・消費税込み￥32,754)……特価￥31,300
④ PIO-6BE4-4M 定価￥88,000
(送料・消費税込み￥56,650)……特価￥54,500

限定 50台限り
■オムロン＝モデム
◉MD-24FP5II(MNP5)
定価￥42,800
▶P&A特価￥23,600
(送料・消費税込み￥25,338)

P&A超低金利クレジットをご利用ください!!

## X68000-XVI

※クレジット表は、送料・消費税込み!!

XVI/XVI-HDセットでお買い上げの方に、もれなくプレゼント!!
①「熱血高校サッカー編(￥8,800)」
②「ダウンタウン熱血物語(￥8,800)」
はもちろん、さらにその上、人気の
イ「ロードス島戦記(￥9,800)」
ロ「パロディウス(￥9,800)」
ハ「生中継68(￥9,800)」
ニ「信長の野望武将風雲録(￥9,800)」
ホ「ELLE(エル)(￥7,800)」
の中のいずれか2本をプレゼント!!

X68000-XVI ▶セットでお買い上げの方に●ディスケット10枚●ジョイカード2ケプレゼント中!!

Ⓐセット:CZ-634C-TN+CZ-606D-TN…定価￥447,800▶特価価格はTEL下さい。

| 12回 | 29,100 | 24回 | 15,400 | 36回 | 10,600 | 48回 | 8,300 | 60回 | 7,000 |
|---|---|---|---|---|---|---|---|---|---|

Ⓑセット:CZ-634C-TN+CZ-614D-TN …定価￥503,000▶特価価格はTEL下さい。

| 12回 | 32,700 | 24回 | 17,200 | 36回 | 11,900 | 48回 | 9,300 | 60回 | 7,800 |
|---|---|---|---|---|---|---|---|---|---|

X68000-XVI-HD ▶セットでお買い上げの方に●ディスケット10枚●ジョイカード2ケプレゼント中!!

Ⓐセット:CZ-644C-TN+CZ-606D-TN…定価￥597,800▶特価価格はTEL下さい。

| 12回 | 38,900 | 24回 | 20,500 | 36回 | 14,200 | 48回 | 11,100 | 60回 | 9,300 |
|---|---|---|---|---|---|---|---|---|---|

Ⓑセット:CZ-644C-TN+CZ-614D-TN …定価￥653,000▶特価価格はTEL下さい。

| 12回 | 42,300 | 24回 | 22,300 | 36回 | 15,500 | 48回 | 12,100 | 60回 | 10,200 |
|---|---|---|---|---|---|---|---|---|---|

※上記のモニターを、CZ-604D(定価￥94,800)、CZ-605D(定価￥115,000)、CU-21HD(定価￥148,000)に変更の場合、TEL下さい。超特価で販売致します。

## X68000シリーズ~P&Aスペシャルセット

(送料￥2,000・消費税別)

注目!!
「P&Aスペシャルセット」にもれなくプレゼント!!
●上記XVI/XVI-HDのプレゼント
①、②+イ~ホの中の2本
+さらにその上、
目にやさしい。
Ⓒ「高性能CRTフィルター(￥19,800)」又は、
Ⓓ「SX-WINDOW. Ver1.1」(￥9,800)
をプレゼント!!

※セットでお買い上げの方に、
●ディスケット10枚
●ジョイカード2個
プレゼント中!!

### SUPER

Ⓐセット:P&A特選セット
■CZ-604C
(本体定価￥348,000)
+
■CZ-606D
(モニター定価￥79,800)
▶P&A超特価 ￥268,000

Ⓑセット
■CZ-604C+CZ-604D
定価￥442,800…▶特価￥275,000

Ⓒセット
■CZ-604C+CZ-607D
定価￥447,800…▶特価￥283,000

Ⓓセット
■CZ-604C+CZ-614D
定価￥483,000…▶特価￥306,000

Ⓔセット
■CZ-604C+CU-21HD
定価￥496,000…▶特価￥313,000

### PRO-II

Ⓐセット:P&A特選セット
■CZ-653C
(本体定価￥285,000)
+
■CZ-606D
(モニター定価￥79,800)
▶P&A超特価 ￥218,000

Ⓑセット
■CZ-653C+CZ-604D
定価￥379,800…▶特価￥225,000

Ⓒセット
■CZ-653C+CZ-607D
定価￥384,800…▶特価￥233,000

Ⓓセット
■CZ-653C+CZ-614D
定価￥420,000…▶特価￥256,000

Ⓔセット
■CZ-653C+CU-21HD
定価￥433,000…▶特価￥263,000

### SUPER-HD

Ⓐセット:P&A厳選セット
■CZ-623C
(本体価格￥498,000)
+
■CZ-606D
(モニター定価￥79,800)
▶P&A超特価 ￥328,000

Ⓑセット
■CZ-623C+CZ-604D
定価￥592,800…▶特価￥335,000

Ⓒセット
■CZ-623C+CZ-607D
定価￥597,800…▶特価￥343,000

Ⓓセット
■CZ-623C+CZ-614D
定価￥633,000…▶特価￥366,000

Ⓔセット
■CZ-623C+CU-21HD
定価￥646,000…▶特価￥373,000

### EXPERII

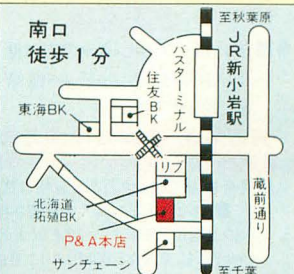
Ⓐセット:P&A厳選セット
■CZ-603C
(本体価格￥338,000)
+
■CZ-606D
(モニター定価￥79,800)
▶P&A超特価 ￥238,000

Ⓑセット
■CZ-603C+CZ-604D
定価￥432,800…▶特価￥243,000

Ⓒセット
■CZ-603C+CZ-607D
定価￥437,800…▶特価￥252,000

Ⓓセット
■CZ-603C+CZ-614D
定価￥473,000…▶特価￥277,000

Ⓔセット
■CZ-603C+CU-21HD
定価￥486,000…▶特価￥280,000

●本広告の掲載の商品の価格については、消費税は含まれておりません。●営業時間＝平日AM10:00~PM7:00、日祭AM10:00~PM6:00

# 全国通販

■アフターサービス万全のサポート体制
●下取・買取は電話で見積りしております。責任を持って下取りさせて頂きます。

営業時間
平日………AM10:00～PM7:00
土日・祭日…AM10:00～PM6:00

SHARP 認定 PPO-SHOP **O.A.ランド**

(TEL) 03-**3770-8855**

▶**12・18～1・17**

SHARPのことなら なんでもおまかせ!!

大徳買セール! 安く値切ってネ。(本体セット:送料 消費税込み)
お電話下さい。㊙価格をお知らせいたします。

流通事情により、広告表示価格は、お安くなる場合がありますので、ドンドンお電話下さい。

CYBER STICK
■CZ-8NJ2 (定価¥23,800)
OAランド特価 ▶¥18,000

電子手帳 ●見やすい漢字4行表示!! 情報化時代の必需品!!
■PA-9500(¥48,000)…▶特価¥38,000
■PA-8500(¥28,000)…▶特価¥15,000
■PA-7500(¥22,000)…▶特価¥12,000

低金利クレジットをご利用下さい。平日AM10時～PM7時、土日・祭日AM10時～PM6時迄ガンバッテま～す!!

## SHARP X68000 XVI

XVIセットでお買い上げの方に

**特典1**
①ディスケット20枚
②連射式JOYパッド
③ゲームソフト2本
④バックアップツールをプレゼント!!

**特典2**
X68000用のゲーム ビジネスソフトと サイバースティックが **30%off**

### ■X68000XVI

①CZ-634C+CZ-614D 定価¥503,000▶現金特価TEL (送料税込)

| 12回 | 24回 | 36回 | 48回 |
|---|---|---|---|
| ¥32,800 | ¥17,400 | ¥12,100 | ¥9,500 |

②CZ-634C+CZ-607D 定価¥467,800▶現金特価TEL (送料税込)

| 12回 | 24回 | 36回 | 48回 |
|---|---|---|---|
| ¥30,700 | ¥16,300 | ¥11,300 | ¥8,900 |

③CZ-634C+CZ-606D 定価¥447,800▶現金特価TEL (送料税込)

| 12回 | 24回 | 36回 | 48回 |
|---|---|---|---|
| ¥29,200 | ¥15,500 | ¥10,800 | ¥8,400 |

### ■X68000XVI-HD

①CZ-644C+CZ-614D 定価¥653,000▶現金特価TEL (送料税込)

| 12回 | 24回 | 36回 | 48回 |
|---|---|---|---|
| ¥42,600 | ¥22,600 | ¥15,700 | ¥12,300 |

②CZ-644C+CZ-607D 定価¥617,800▶現金特価TEL (送料税込)

| 12回 | 24回 | 36回 | 48回 |
|---|---|---|---|
| ¥40,600 | ¥21,400 | ¥14,900 | ¥11,700 |

③CZ-644C+CZ-606D 定価¥597,800▶現金特価TEL (送料税込)

| 12回 | 24回 | 36回 | 48回 |
|---|---|---|---|
| ¥39,000 | ¥20,700 | ¥14,400 | ¥11,300 |

### XVI+HDD限定セット

①CZ-634C-TN
CZ-606D-TN
Curent-80FX
31%off
定価合計¥555,800
**特価¥386,000**

②CZ-634C-TN
CZ-614D-TN
TX-130B
31%off
定価合計¥641,000
**特価¥441,000**

そして……
在庫処分3台限り

③CZ-604C-TN
定価¥348,000
49%off
**特価¥179,000**

■IO-735X-B 定価¥248,000
**特価¥155,000**

■ゲームソフト、ビジネスソフト2割、3割引きは、当り前で～す。お問い合せ下さい。

### X68000周辺機器

●CZ-6VT1 ……特価¥ 51,000
●CZ-6TU ……特価¥ 24,800
●CZ-8NS1 ……特価¥134,500
●JX-220X ……特価¥120,000
●CZ-6BNI ……特価¥ 22,000
●CZ-6BMIA ……特価¥ 19,800
●CZ-6BCI ……特価¥ 57,000
●CZ-6BGI ……特価¥ 43,000
●CZ-6BP1 ……特価¥ 57,000
●CZ-6BP2 ……特価¥ 33,000
●CZ-6BFI ……特価¥ 36,000
●CZ-6EBI ……特価¥ 63,500

### ハードディスク

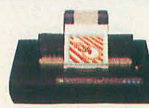
TX-180B
定価¥185,000
特価¥122,000

■Itec
●TX-80B (定価¥108,000) ……特価TEL下さい。
●TX-130B (定価¥138,000) ……特価¥ 91,000

■日本アルトス社
●Curent-80FX ……特価¥ 75,000

■SHARP
●CZ-64H (定価¥120,000) ……特価¥ 86,000
●CZ-68H (定価¥160,000) ……特価¥115,000
●CZ-6M01 (定価¥450,000) ……特価¥322,000
●メディア (定価¥30,000) ……特価¥ 25,000

※SCSIボード
●CZ-6BS1 (定価¥29,800) ……特価¥ 22,000

### プリンター

IO-735X-B
定価¥248,000
ケーブル/IO-73CX付
特価¥169,500

CZ-8PC5
定価¥96,800
大特価TEL下さい。

●CZ-6PV1 (定価¥198,000) ……特価¥142,000
●CZ-8PG1 (定価¥130,000) ……特価¥ 92,800
●CZ-8PG2 (定価¥160,000) ……特価¥114,000
●CZ-8PK10 (定価¥ 97,800) 特価¥ 69,900

☆すべて用紙とケーブルが付いてます。

### RAMボード

■計測技研(増設メモリとコプロが1つに!!)
●KGB-X68PRKII-02 (定価¥ 55,000) ……特価¥ 41,000
● PRKII-04 (定価¥ 90,000) ……特価¥ 68,000
● PRKII-06 (定価¥125,000) ……特価¥ 94,000
● PRKII-08 (定価¥160,000) ……特価¥121,000
● PRKII-12 (定価¥ 85,000) ……特価¥ 64,000
● PRKII-14 (定価¥120,000) ……特価¥ 91,000
● PRKII-16 (定価¥155,000) ……特価¥116,000
● PRKII-18 (定価¥190,000) ……特価¥143,000
●MC-68881RC (定価¥ 38,000) ……特価¥ 27,800

■I/Oデータ
●PIO-6BE1-A (定価¥25,000) ……特価¥16,000
●PIO-6BE2-2M (定価¥50,000) ……特価¥31,500
●PIO-6BE4-4M (定価¥88,000) ……特価¥55,000

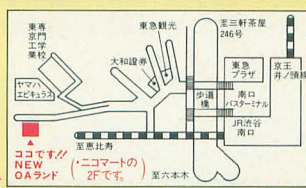
■SHARP
●CZ-6BE1 (定価¥35,000) ……特価¥25,800
●CZ-6BE2A (定価¥59,800) ……特価¥43,000
●CZ-6BE2B (定価¥54,800) ……特価¥39,800

### 通信販売のご案内

全国通販

■銀行振込で申し込みの方は商品名及びお客様の住所・氏名・電話番号をお知らせ下さい。
〔振込先〕第一勧業銀行 東新宿支店 普通No.1051605 ㈱オーエーランド

■年中無休です!!

■現金書留で送金されるお客様は電話番号と商品名、数量を明記して同封して下さい。■クレジットでご購入を希望される方は申し込み用紙をお送り致しますのでご記入の上返送して下さい。20才以上の方は、原則として保証人不要です。クレジットは1～60回払で月々5,000円よりご自由に設定できます。

クレジット表

| 3回 | 3.5% | 6回 | 4.5% | 10回 | 6% | 12回 | 6% | 15回 | 8.5% | 18回 | 11% | 20回 | 12% |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 24回 | 12.5% | 30回 | 17% | 36回 | 17.5% | 42回 | 22.5% | 48回 | 23% | 54回 | 29% | 60回 | 29.5% |

**㈱オー・エー・ランド**
〒150 東京都渋谷区桜丘町3-13 アルカディア2F
☎(03)3770-8855
関東エリアの送料は、1個につき¥1,000です。 FAX(03)3770-7080

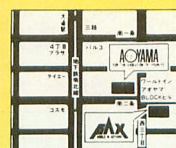
★全商品保証書付。専門のアドバイザーが、お客様のニーズに対応します。
★初期不良・輸送トラブル等に迅速に対応し、即交換させていただきます。

■本体・モニターのセットは、すべて送料・消費税込です。掲載の価格は、11月下旬現在です。

AOYAMA NEWイヤーズフェア12 26〜1 15全店にて初売セールを開催。

面白商品を破格値でご奉仕します。掘出し物がキットある！お正月は1 3、12時より全店同時初売市開始。

「各店のお休み」 1月のお休み／9日(木)・16日(木)・23日(木)・30日(木)

**池袋本店**
豊島区東池袋1-28-1
■営業時間／11:00〜19:00

**札幌店** 札幌市中央区南2条西3丁目 リンクエギビル3F ■営業時間／10:00〜19:30
**札幌AAX店** 札幌市中央区南2条西2丁目 ブロックビル6F ■営業時間／11:00〜19:30

電話でのご注文の場合 (03)3987-7771 ●北海道受注センター 011-251-6771 ●九州受注センター 092-716-7771 〈取引銀行〉住友銀行池袋支店 〈口座番号〉普通 1065392

## SHARP X68000

### X68000万全のサポート

AOYAMAにて購入のX68000は万一故障の場合でも全国どこでも出張サービスがうかがいます。 万一の場合ワールドインアオヤマサポート係にお電話下さい。お客様のお名前と電話番号だけで手続きは完了。

**CZ-634C-TN**
- CZ-634C-TN(2M本体16MHz) ¥368,000
- CZ-607D-TN(.31 14インチチューナー付) ¥99,800
- 3Mフロッピーディスケット ¥9,000

定価合計¥476,800➡現金特価
クレジットは、お電話にて御問い合わせください。

**X68000 CZ-634C-TN**
- CZ-634C-TN(2M本体16MHz) ¥368,000
- CZ-614D-TN(.31 15インチチューナー付) ¥135,000
- 3M フロッピーディスケット ¥9,000

定価合計¥512,000➡現金特価
クレジットは、お電話にて御問い合わせください

**CZ-634C-TN**
- CZ-634C-TN ¥368,000
- CZ-606D-TN ¥79,800
- AP-900 ¥92,800
- プリンターケーブル ¥4,800

定価合計¥550,400➡¥369,800
クレジットは、お電話にて御問い合わせください。

**CZ-644C-TN**
- CZ-644C-TN(2M本体16MHz80MHDD) ¥518,000
- CZ-614D-TN(.31 15インチチューナー付) ¥135,000
- 3Mフロッピーディスケット ¥9,000

定価合計¥662,000➡現金特価
クレジットは、お電話にて御問い合わせください。

**SX68M**
- SX68M ¥19,800
- CM32L ¥69,000
- MA12AV×2 ¥28,000

定価合計¥117,600➡¥94,000
クレジットは、お電話にて御問い合わせください。

**CZ-653C-BK / CZ-606D-BK** ¥218,000
- CZ-653C-BK(1M本体) ¥285,000
- CZ-606D-BK(.31 14インチカラーCRT) ¥79,800

定価合計¥382,400➡¥226,000

**CZ-634C-TN / CZ-603D** ¥299,000
- CZ-634C-TN ¥368,000
- CZ-603D(限定12台) ¥79,800

定価合計¥447,800➡¥299,000

**CZ-604C-TN / CZ-606D-TN** ¥268,000
- CZ-604C-TN ¥348,000
- CZ-606D-TN ¥79,800
- 住友3M 5'2HD サービス品

定価合計¥427,800➡¥268,000

**CZ-623C-TN / CZ-606D-TN** ~~¥299,000~~
- CZ-623C-TN ¥498,000
- CZ-606D-TN ¥79,800
- CZ-8NJ2＋ソフトサービス

定価合計¥577,800➡¥328,000

**SHD-40** 特価 ¥62,000

**5台限り TX-130** 特価 ¥96,800

**SHD-80** 特価 ¥95,000

**TX-180** 特価 ¥185,000

### X68000ソフト＆周辺機器

X68000をはじめソフト＆周辺機器類は、当社池袋店・札幌店・旭川店・福岡店にて実演中です。各店X68000コーナーが常設されております。

| 商品 | 内容 | 価格 | 商品 | 内容 | 価格 | 商品 | 内容 | 価格 |
|---|---|---|---|---|---|---|---|---|
| システムサコムSX-68M | MIDIボード | ¥19,800➡¥15,250 | SHARP IO-735X | 136桁インクジェットプリンタ | ¥248,000➡¥168,000 | ローランド MT-32 | MIDI音源 | ¥64,000➡¥49,800 |
| アイテック TX-80 | 80MB HDD | ¥108,000➡¥80,000 | アイテック TX-130 | 130MB HDD | ¥138,000➡¥111,000 | SHARP CZ-8PK10 | 136桁ドットプリンター | ¥97,800➡¥70,000 |
| I/Oデータ PIO-6BE1A | IMB増設RAM | ¥25,000➡¥17,800 | ハル研 HGS-68 | ファインスキャナー68 | ¥39,800➡¥29,800 | SHARP BF-68PRO | テレビフィルター | ¥19,800➡¥14,800 |
| SHARP マルチワード | マルチワープロソフト | ¥32,000➡¥24,000 | SHARP CZ-6BE1 | CZ-600C専用IMB増設RAM | ¥35,000➡¥26,800 | SHARP CZ-6BM1 | MIDIボード | ¥26,800➡¥19,800 |
| SHARP Ccompiler PRO-68K | Cコンパイラ | ¥44,800➡¥33,600 | SHARP CZ-6BE1B | IMB増設RAM | ¥28,000➡¥21,800 | アイテム X Stor40 | HDD | ¥118,000➡¥89,800 |
| システムサコム Mu-1 Suger | MIDI用ソフト | ¥39,800➡¥29,800 | SHARP JX-220XB | イメージスキャナ | ¥168,000➡¥134,400 | | | |
| SHARP CZ-8PC5 | 80桁熱転写プリンタ | ¥94,800➡¥69,800 | SHARP CZ-8NJ2 | インテリジェントコントローラー | ¥23,800➡¥18,800 | | | |

全国出張サポート★私共にてご購入いただいたX68000は全国出張サポートがうけられます。

**超お買得品**(本体・ディスプレイ・プリンター・周辺) 詳しくは 03-3987-7771

## 実績のメンテナンス／完全バックアップシステム

#### クレジット

**業界一番のスーパークレジットで。**

ご都合に合せて自由にお支払い回数、金額を設定できます。手続きはとても簡単、そしてスピーディー。もちろん手数料は超低金利。お支払いも月々1,000円からとますますお得に使いやすくなったアオヤマのクレジットです。

**学生の味方、キャンパスクレジットがますますワイドに。**

キャンパスクレジットのワクがグーンと拡がりました。最高50万円までのお買物ができ、手続きもその場で即OK。お支払いも月々1,000円からととっても嬉しいアオヤマは皆さんを応援します。

**お支払いはナント！84回まで。**

お支払いは少なく、ゆっくりと。ワールドインアオヤマなら最長84回まで自由にセッティング。ボーナス額を増やしたり減らしたりはもちろん、ボーナス一括や二括払いもOK。

**ゆうゆうお支払いは8ヶ月先から。**

欲しい機種は今すぐ欲しい、だけど以前のローンが残ってるし……。就職してから始めたい……何んて方にお勧め。8ヶ月先までならいつから始めてもOKです。これならとっても安心ですね

#### サポート

**万一のときも完全バックアップ。**

万一の初期不良があった場合でも当社では万全の体製でお客様をフォロー致します 通常の初期不良ワクを大きく拡げ、最長1ヶ月間まで新品との交換を致しております

また一週間以内の不良の場合は、こちらからお荷物をひきとりに伺います

※ソフト及び中古商品に関しましては1ヶ月サービスの対象外とさせて頂きます

**グーンとお得な下取りシステム。**

今お持ちの機種を高額下取り わずかな予算でシステムアップ 差額はクレジットでもOKです

#### ショップ

**全店統一のサービス／見て、触れて、納得。**

ショールームでは、お客様に一切声をかけないこともサービスのひとつと思っております ズラリ勢揃いしたハード＆ソフトの中から、ご自身の目でじっくりとお選び下さい

**BEST SHOPS BEST SERVISE。**

私共では、お客様の過去のデータ、その日の在庫量、価格など全てのデータが完全管理 だから、アフターサポートも電話一本で

#### サービス＆ポリシー

**グーンとお得な下取りシステム。**

**電話でのご注文の場合**

03-3987-7771

お電話番号はおかけ間違いのないようにお願いします。

北海道受注センター 011-251-6771
九州受注センター 092-716-7771
お好きな時間にお電話を！

**クレジットカードもOK！**

クレジットカードをお持ちの方お支払は1回払いです。お申し込みの際、①カード名②会員No.③有効期限をご連絡下さい。

カードでお申し込みの場合、販売価格か変わりますのでお電話にてお問い合せ下さい

**お客様相談室**

03-3987-7795

すでにご注文いただいているお届け時間(時期)やメンテナンス、その他のお問い合せは上記へお電話下さい。

**ファクシミリでご利用の場合**

03-3985-5221

●ご注文方法(黒色のボールペン、またはサインペンでご記入下さい。)
①電話番号・住所・氏名又はお客様番号、お支払い方法をご記入下さい

●銀行振込みの場合

| | |
|---|---|
| 取引銀行 | 住友銀行 池袋支店 |
| 口座番号 | 普通 1065392 |
| 口座名 | 株式会社 ワールドイン アオヤマ |

返品・交換についてのお問い合わせは お客様相談室 03-3987-7795

X68000及びFM-TOWNS全国出張サポート

私共にて購入いただきましたX68000及びFM-TOWNSはどこでも出張サポートが受けられます。 シャープX PRO SHOP・富士通 FM-SHOP会加入のプロショップです。

アオヤマのクレジット金利は60回でも29.5%とたいへんオトク!! 現金でもクレジットでもお受けしております。

信頼と安さの TSUKUMO

掲載商品2万円以上送料無料(離島を除く)

# 決算迄の FINAL DASH SALE.

**毎年恒例!秋葉原電気まつり**

賞金総額7,000万円　11/22金～1/7火

ますます好評の秋葉原電気まつり。秋葉原の店頭でお買物をされたお客様、商品代金¥5,000毎に抽選券を1枚さし上げます。

- 1等 =現金10万円
- 2等 =現金5万円
- 3等 =現金3万円
- 4等 =現金5千円

FINAL DASH ディスケットプレゼントセール　12月31日迄

期間中にツクモTSシリーズのTSドライブ(商品代金2万円以上)お買い上げのお客様にディスケット3.5インチ=5枚or5インチ=10枚をプレゼント!!

新製品の情報もいろいろな機種の情報も、ツクモに行けばあなたのもの。歩きまわる必要はありません。ツクモだけで十分です!!

★★★★★★★★★★シャープ製品は、小さいモノはポケコンから大きいモノは液晶ビジョンまで、何でも揃う!★★★★★★★★★★

大好評につき在庫お問合せ下さい。

## ツクモX68000用TSドライブ

「…►目のつけどころがツクモでしょ。」

X68000シリーズ用3.5インチフロッピーディスクドライブ

●3.5インチ2DD/2HD対応ドライブ使用。●2DD用ディバイスドライバ付属。●1.44MBディバイスドライバ付属。

3.5インチ1ドライブ **TS-3XR1**
定価¥44,800
ツクモ特価¥35,800
(消費税別途¥1,074)

3.5インチ2ドライブ **TS-3XR2**
定価¥57,800
ツクモ特価¥46,800
(消費税別途¥1,404)

超低金利 冬・夏ボーナス2回払受付中!! 詳しくは☎03(3251)9911、又は各店へ

## X68000用ハードディスク

大容量記憶装置

130MB SCSI対応タイプ
**TX-130** 定価¥138,000 ツクモ特価¥96,000
SCSIボードセット¥120,000

180MB SCSI対応タイプ
**TX-180** 定価¥185,000 ツクモ特価¥130,000
SCSIボードセット¥154,000

SCSIタイプのHDDの場合、本体がSUPER/XVI以外の場合にはSCSIボード(CZ-6BS1)が必要です。

## これがなければ始まらない!!

## X68000 XVI 快速16MHz

●CPUクロック周波数スピードアップ(16MHz)●増設メモリ本体内蔵可能(8MBまで)●NEW SX-WINDOW可能。

■X68000XVI (CZ-634C-TN)
標準タイプ……¥368,000

■X68000XVI-HD (CZ-644C-TN)
HDD内蔵タイプ¥518,000

### お買得ハードディスクセット

- ●CZ-634C-TN……¥368,000
- ●CZ-614D-TN……¥135,000
- ●TX-130……¥138,000

合計定価¥641,000

ツクモ特価¥489,000
(消費税別途¥14,670)

クレジット例(48回払・税込)
初回¥17,7??+月々¥13,500×47回

(買い換え・下取りも取り扱っております 是非、お尋ね下さい)

ツクモニューセンター店 ☎03-3251-0987

### X68000シリーズ用オプションボード

- ■1MB増設RAMボード(CZ-600C専用)…特価¥20,000
- ■1MB増設RAMボード(ACE/PRO/PRO2シリーズ用)…特価¥17,500
- ■2MB増設RAMボード(拡張スロット専用)…特価¥34,800
- ■4MB増設RAMボード(拡張スロット専用)…特価¥61,500
- ■GP-IBボード……特価¥50,800
- ■増設RS-232Cボード……特価¥42,300
- ■スキャナ用パラレルボード……特価¥25,300
- ■ユニバーサルI/Oボード……特価¥33,800

※計測技研のメモリーボードも取扱っておりますので、価格についてはお尋ね下さい。

### 大容量が欲しい方に、ツクモはSONY MOの認定店です。

新製品

- ●RMO-S350……¥235,000
- ●SCSIケーブル……¥6,900
- ●光磁気カートリッジ……¥7,000
- ●SCSIインターフェースボード……¥29,800

合計定価¥278,700

ツクモ特価¥238,000

シャープ純正「CZ-6MO1」も特価販売中!

## コンピュータミュージック(X68000用)

**NEW Aセット**
- ●CM-32L……¥69,800
- ●SX-68M-II……¥19,800
- ●Musicstudio Mu-1 Ver1.4……¥19,800

合計定価¥109,400
ツクモ特価¥88,000
(消費税別途¥2,640)
クレジット例(18回払・税込)
初回¥7,223+月々¥5,600×17回

**NEW Bセット**
- ●CM-300……¥58,000
- ●SX-68M-II……¥19,800
- ●Mu-1 SUPER……¥39,800

合計定価¥117,600
ツクモ特価¥92,000
(消費税別途¥2,760)
クレジット例(10回払・税込)
初回¥10,967+月々¥10,100×9回

**NEW Cセット**
- ●CM-500……¥115,000
- ●SX-68M-II……¥19,800
- ●Mu-1 SUPER……¥39,800

合計定価¥174,600
ツクモ特価¥141,000
(消費税別途¥4,230)
クレジット例(15回払・税込)
初回¥17,079+月々¥10,600×14回

**NEW Dセット**
- ●CM-64……¥129,000
- ●SX-68M-II……¥19,800
- ●Mu-1 SUPER……¥39,800

合計定価¥188,600
ツクモ特価¥154,000
(消費税別途¥4,620)
クレジット例(18回払・税込)
初回¥10,940+月々¥9,900×17回

※この他の組み合わせは、お問い合わせ下さい。☎03-3251-9911へ

**ローランド 追加オプション機器**
- ステレオマイクロモニター CS-10……定価¥17,000
- MIDIキーボードコントローラー PC-200……定価¥36,000

※本格的MIDIは7号店2F MIDIフロア☎03-3253-4199へ

## X68000用ならなんでも揃っています。

### アートツール(ハード)
- ■JX-220X A4サイズカラーイメージスキャナー……定価¥168,000
- ■HGS-68 ファインスキャナーX68……ツクモ特価¥29,800

### アートツール(ソフト)
- ■CANVAS PRO-68K…定価¥29,800
- ■NEW PrintShop PRO-68K Ver.2……定価¥20,000
- ■Z's STAFF PRO-68K Ver.2……ツクモ特価¥46,400
- ■マジックパレット……ツクモ特価¥15,800

### 開発ツール
- ■C Compiler PRO-68K Ver2.0……定価¥44,800
- ■XBAS TO C CHECKER PRO-68K……定価¥9,800

### SX-Windowツール
- ■SX-Window Ver1.1 定価¥9,800
- ■EasyPaint SX-68K 定価¥12,800
- ■SX-Windowイラスト集 Vol.1 (一般実用編)……定価¥8,000
- ■SX-Windowイラスト集Vol.2 (行事・四季編)……定価¥8,000
- ■SOUND SX-68K…近日発売予定
- ■Communication SX-68K……近日発売予定

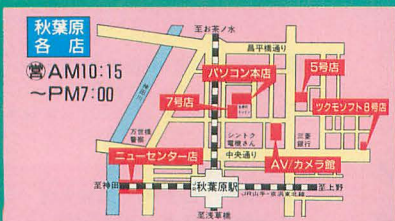
### パソコン通信
- ■モデム 2400ボー/MNP5&V42bis対応…ツクモ特価¥29,800
- ■通信ソフト た～みのる2……ツクモ特価¥14,000
- ■Telepotion PRO-68K……定価¥22,800

### ビジネスツール
- ■Press Conductor 定価¥28,000
- ■Multiword……定価¥32,000
- ■FIXER Ver4.0…特価¥15,800
- ■CARD PRO-68K Ver2.0……定価¥29,800
- ■CARD PRO-68K Ver2.0 パーソナルプログラム集…定価¥12,000
- ■CARD PRO-68K Ver2.0 ビジネスプログラム集……定価¥12,000
- ■CHART PRO-68K 近日発売予定

### 電子手帳
- ■ハイパー電子システム手帳
  - PA-9500……定価¥48,000 ツクモ特価¥43,000
  - PA-9550……定価¥59,000 ツクモ特価¥53,000
- ■スタンダードタイプ電子システム手帳
  - PA-S1……定価¥22,000 ツクモ特価¥19,800

**商品のご注文は通販受注専用センター** フリーダイヤル **0120-377-999**

商品についての詳しいお問い合わせは各店、又は☎03(3251)9911へ

秋葉原各店
営AM10:15～PM7:00

ツクモは「スーパーX PRO SHOP」です。

PRO STAFF ツクモ

九十九電機株
〒101-91 東京都千代田区神田郵便局私書箱135号
★商品のご注文は在庫確認の上お願いします。★表示価格には消費税は含まれておりません。

- ■ツクモパソコン本店2F ☎03-3253-5599(担当 荒井) 休毎週木曜
- ■ツクモニューセンター店 ☎03-3251-0987(担当 福地) 休毎週木曜
  (12月は無休・正月は1/2から但しニューセンター店は1/4から)

**便利で安心な通信販売**
**ツクモ通販センター☎03-3251-9911**

- ■ツクモ5号店 ☎03-3251-0531(担当 森) 休毎週木曜 (12月は無休・正月は1/4から)
- ■ツクモAV/カメラ館B1 ☎03-3254-3999(担当 川名) 休毎週水曜 (12月は無休・正月は1/2から)
- ■名古屋1号店 ☎052-263-1655(担当 吉高) 休毎週月曜 (12/12より1/8迄無休)
- ■名古屋2号店 ☎052-251-3399(担当 横山) 休毎週水曜 (12/12より1/8迄無休)
- ■ツクモ札幌店 ☎011-241-2299(担当 田口) 休毎週木曜 (12月は無休・正月は1/2から)

安心 迅速 高額
ツクモニューセンター店
買い取りセンター好評買い取り中
電話受付 (AM11:00～PM5:00)
☎03(3251)9977

**ツクモグローバルカード**
大人気/入会者募集中!
国内・外で活躍!使って便利、持ってて安心!ツクモグローバルカードはジャックス・VISAとの提携カードです。ツクモ各店でのお買物がらくらくできるうえに、国内はもとより海外での分割ショッピングもOK!
お申し込みは☎03(3251)9898又は店頭にて!
18才以上なら学生でもOK。
※各店頭では、JCB・日本信販・DC・セントラル・マスター他各種カードも取り扱っております。

| カード払い | 全国代金引き換え配達 | クレジット払い | 現金書留払い | 銀行振込払い | 各種リース払い |
|---|---|---|---|---|---|
| 通信販売での御利用カード、ツクモグローバルカード、VIPカードセントラル、ジャックス※御本人様より電話で通信販売部へお申し込み下さい。 | お申し込みは☎03-3251-9911へお電話1本 配達日の指定もできます。 | 月々¥3,000以上の均等払いも頭金なし。夏・冬ボーナス2回払いも受付中!! | 〒101-91 東京都千代田区神田郵便局私書箱135号 ツクモ通販センター oh!X係 | 事前に☎でお届け先をご連絡下さい。三和銀行 秋葉原支店 (普)1009939 ツクモデンキ | くわしくは各店にお問い合わせ下さい。ケースに合わせてご相談にのります! |

◆◆◆◆企業の方へ…お見積りはFAXで。ツクモパソコン本店FAX03-3253-5199担当/荒井へ◆◆◆◆

[第8回]
年賀状

寺尾　響子

# 響子inCGわ〜るど

「まだかな，まだかな」

やわらかな日ざしがガラス戸ごしに入ってきます。厚手のセーターを着込んだ背中に太陽の光があたってじんわりと暖かい。お腹はいま食べたお雑煮でいっぱい。猫のように背をまるくして，こたつに入ってうとうととまどろんでいます。

「まだかな」

結婚をして遠くへ行ってしまった高校のときの友達はどうしているかしら。去年の年賀状の写真は新婚旅行の風景でした。赤ちゃんができているかも。会いたいな。

Fさんは毎年墨一色の木版画の年賀状。メッセージが必ずひとこと添えてあります。今年はどんなのか楽しみ。いつでも会える人だけれど，はがきをもらうとなんだか嬉しい感じがします。

かわいい干支のキャラクターをラインマーカーで描いていたCさんが，留学先のアメリカで交通事故に遭い亡くなったのは……あれは何年前のこと？　少し黄ばんできたその年賀状はいまでも大切にとっておいてあります。

「まだかな」

私の年賀状づくり。12年前の干支の猿は，1枚1枚ペンで描いていました。プリントごっこを使ったことも，シルクスクリーンでつくったことも，コピーで簡単にすませたこともあります。年を経るごとに人づきあいが増えて，出す年賀状の枚数も多くなってきました。CGでつくりはじめたのは，去年の羊年から。そして今年の猿も。CGのRGBデータが入ったフロッピーを現像所に持ってゆきます。ポジ出力をしてネガにいったん変えて，あとはふつうの写真年賀状とおなじように焼き増しして出来上がり。プリンタは使いません。いまのところこの方法がいちばん気に入っていて，来年もそうしてつくろうかと思っています[1]。でも，12年後にまた猿年がめぐってきたとき，私はいったい何を使って年賀状をつくっているかしら。

3次元空間に

きもちを　こめて

届けたい

あなたに

---

1）フジカラーサービスのCGハードコピーサービスを利用します。RGBデータを高精度フィルムレコーダでポジフィルムに出力します。写真年賀状にする場合には，このポジをいったんネガに起こしてから使います。

# Oh!X Graphic Gallery

## DoGA・CGアニメーション講座

今月のCGAは，かまた氏作の「GIFTED」です。才能がありながら弱気なボクサーロボット「トロル」が，初めての試合に臨むところを描いた作品です(詳しいストーリーは本文を参照のこと)。また，各々のグラフィックの下にリスト3の絵コンテの番号を明記してありますので，参考にしてください。
右下の作品は，かまた氏が「X68000芸術祭」に出展した「EYE」です。こちらも本文と併せてご覧ください。

❶ A1

❷ A2T1

❸ B1T2

❹ C1T2

❺ D1T1

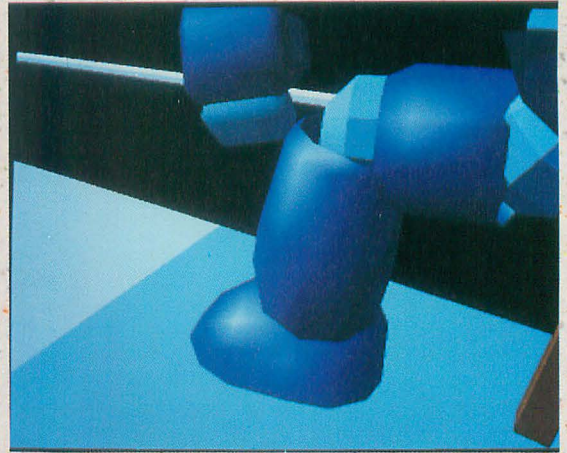
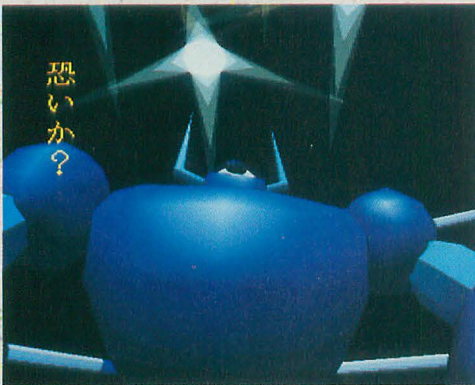
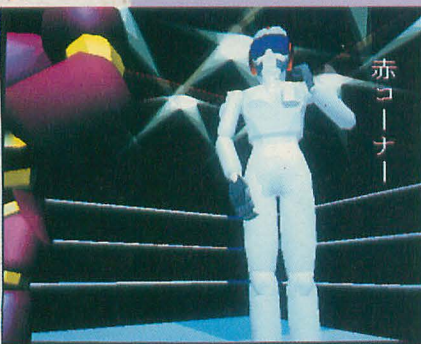
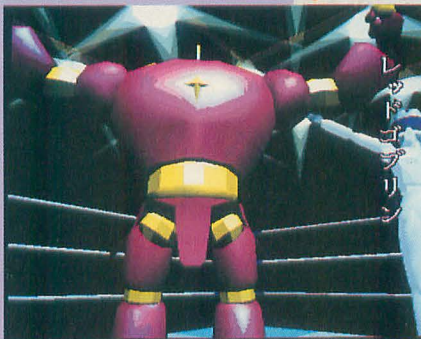
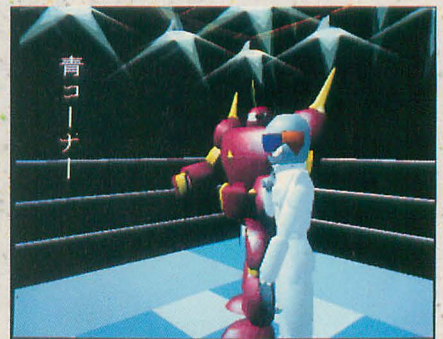
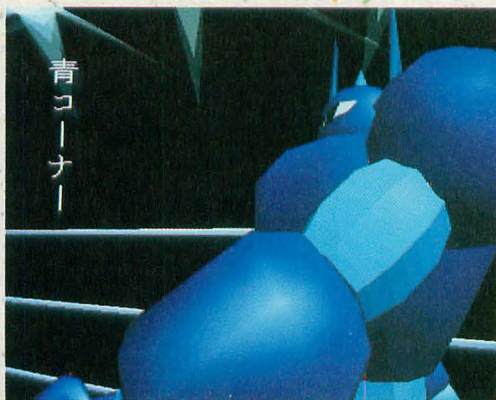
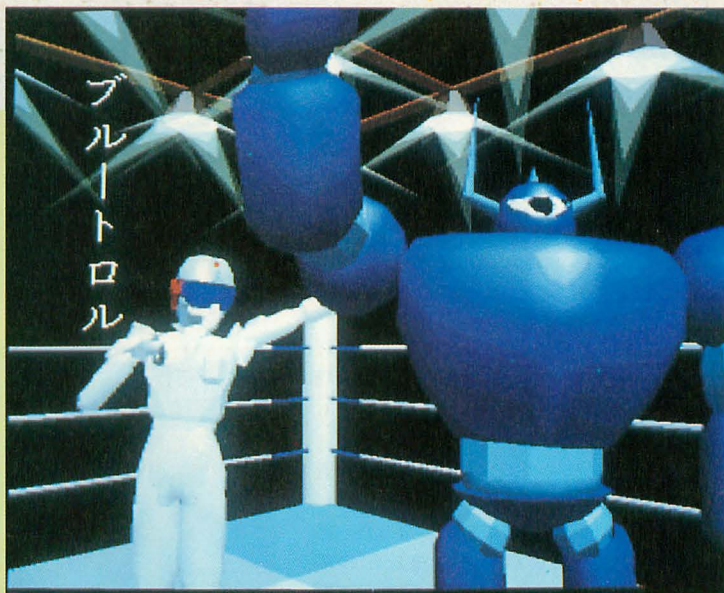
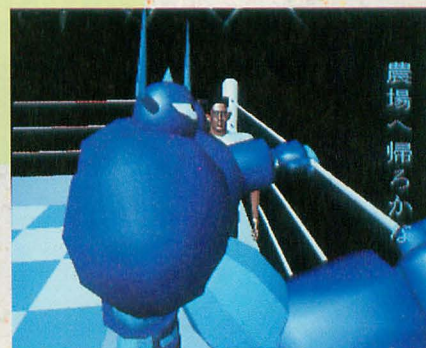
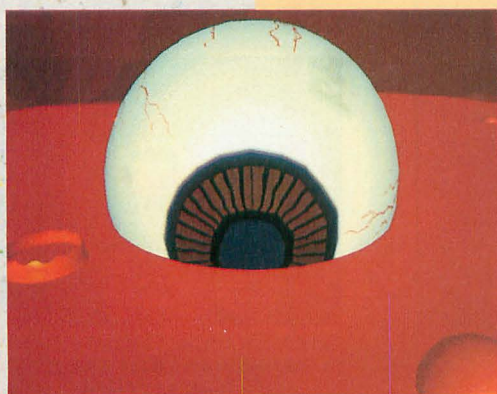
❻ D3

❼ E2T1

❽ F1

❾ G4

❿ H1T1

⓫ I1T4

⓬ J3T1

いったいこの目玉のお化けはなんなのか!? そしてこの青い怪物は？ その謎は芸術祭本選で明らかになる（らしい）。

# 第1回全日本X68000 芸術祭 首都圏大会

まずは選手宣誓から

大盛況の人の山と芸術祭のタレ幕

X68000芸術祭の首都圏大会が11月24日に行われた。予想以上の来場者数を迎え，本会場のほかにプロジェクタで本会場の様子を映し出す，というサブ会場まで用意されるほど。本来は10作品のはずのエントリー作品も，近畿地区大会と同様に15作品が選出され，レベルの高い作品どうしの競い合いとなった。結局，大賞に輝いたのは竹内久徳氏制作の「PENGUINランドネットOFF会記念」。顔写真を取り込んで，シューティングの敵キャラとしたもので，攻撃方法も面白かった。これでX68000芸術祭も，九州大会，補選を残すのみとなり，大詰めを迎える。

こちらはサブ会場

PC-8801MCが目をひいた

この作品が大賞となった

ウワサのPress Conductor PRO-68K

88の「ダンスディスク」も入選

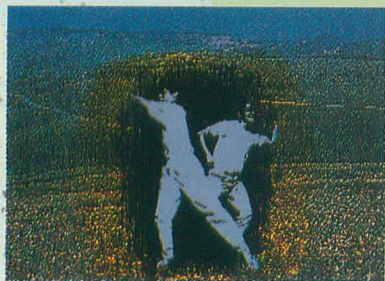
投票タイムも大にぎわい

## ほかの地区の結果は？

首都圏大会の前に，中部，北陸，近畿の各地区大会も行われている。どの地区大会も盛り上がりを見せたようだが，ここでは大賞受賞作品を簡単に紹介しておこう。

中部地区大賞「見上げてごらん」。寺田光太郎氏制作。好きだった女の子に贈る音楽作品。

北陸地区大賞「FORTRESS ATTACK」。柴原章宏氏制作。横スクロールシューティングゲーム。

近畿地区大賞「RUSH！」。京大マイコンクラブ制作。X68000最大4台を接続する対戦型ゲーム。

## 荻窪圭も行ってきたぞ

なにやら面白そうだというんで，新しい才能を見に（そういう大げさな気分だったのだ），X68000芸術祭の首都圏大会へ行ってきた。なにやら大盛況で，当初用意した会場に全員が収まりきらず，サブ会場までできたほどだった。これで作品がつまんなかったら，かあいそーだなあと思いつつも，これがまたあんまり面白くなかった。だいたい，あたしゃ，「笑いにいったんじゃないぞー」といいたいね。面白かったのは一発ギャグものだけ。

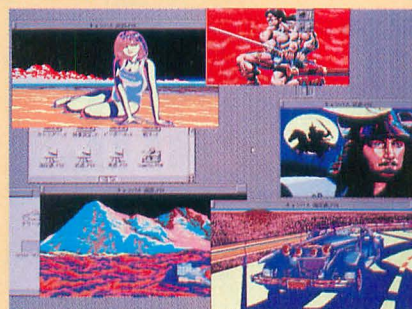
あ，ひとつだけ，評価に値する作品があった。唯一のPC-8801による作品だ。エビ天ネタだったのだが，かけた手間暇といい，表現といい，けっこう傑作だ。踊っている自分たちをコマ撮りしてスキャナで取り込んで，音楽に合わせて表示しただけなのだが，そこには工夫があった。少なくとも，見せる作品にはなっていたぞ。会場ではお笑い作品ととられていたようだが，それでは心の目が曇る，ってものだ（そこまでいうほどのものでもないが）。

X68000の作品にはシューティングゲーム禁止令とグラデーション禁止令を出したい気分である。技術のあるやつはこれといってアイデアもコンセプトもないし，アイデアがあるやつは詰めが甘い。グランプリは当然あの88ユーザーか，と思いきや違った（アイデア一発モノだった）。うーむ。88ユーザーさんにはOh!X賞が送られたそうで（レディースサンタセットだ！），なによりである。さっさとPC-8801をX68000に買い替えて，もっと面白いものを作るように。

# SX-WINDOW 高解像度の世界

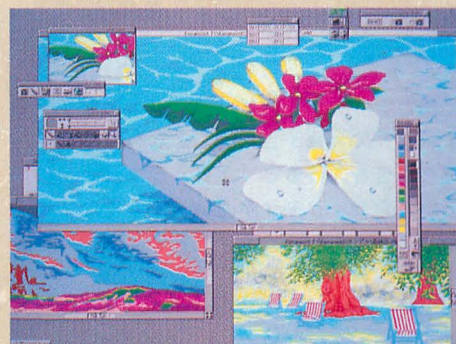
65536色といわれても違和感のないグラフィック表示は縮小されているからではない。高解像度と16色化のアルゴリズム次第でフルカラー画像も再現可能だ。

この広さは魅力だ

メーカー標準のアプリケーション群

24, 16, 12ドットの文字表示

EasyPaint画面

テキスト4階調の例1

テキスト4階調の例2

## ウィンドウがいっぱい

ウィンドウ環境に高解像度は必須。ここでは24kHzインタレースモードを使用して高解像度表示を行ってみた。使用したディスプレイはエプソンの長残光ディスプレイCR-5500。表示されるドット数は1024×848ドット。画面がもう少し大きければまさにワークステーションの雰囲気。ウィンドウ環境を快適に使うにはこれくらいほしいところだが……。

ちなみにCR-5500の値段は168,000円。標準ディスプレイが135,000円とするとちょっと割高か。15kHz〜31kHzまで対応できるので，テレビが見れないことを除けばX68000用として常用しても機能的に問題はない。蛍光体が超残光仕様のため，動きの速い画像は尾を引くのが気になるくらい。17型のCR-7000もあるがこっちはもうちょっと高い。「情報量が勝負を決めるときがある」というエプソンの謳い文句。しかし，同社新鋭のハイレゾ機の表示でも1画面840000ドット，X68000では旧型機でも868352ドットまで可能だ。メガディスプレイの世界まであと200ライン。これだけ低価格に実現できるのだから，使わないともったいない？

16色表示のグラフィック（SXIMAGE.Xによる）を見てほしい。これくらいの解像度があれば，16色だって捨てたものではないだろう。単なるグラフィック表示ならば65536色と比べてもほとんど遜色ない画像を再現できる。

なお，表示されているウィンドウ中には，一部未公開のアプリケーションも含まれている。今回は詳しくは解説しないが，ここに現れているもの以外にもいろいろと「鋭意開発中」のものもある。

さらに，すでにたくさんのソフトウェアがユーザーの手によって作成されている。大規模なものはないが，環境改善のためのツールやゲーム，環境ソフトなどはあちこちのパソコン通信ネット上にアップロードされているようだ。いろいろなアプリケーションを動かす「場所」として，SX-WINDOWは面白い位置を占めているように思われる。

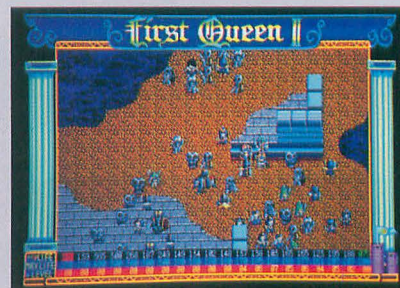
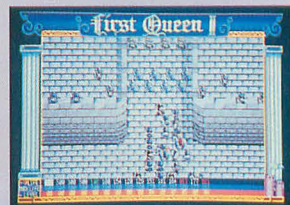
写真のアプリケーションもいずれ公開されると思うので，ほどほどに期待してほしい。

# SOFTWARE INFORMATION

「出たな!! ツインビー」,「ジェノサイド2」などの大物が目白押しとなった昨今。海外に目を向けると「ポピュラスII」がいよいよ発売。IIモノがたいへん多くなっているようですねえ。で，IIといえば……。

## ファーストクィーンII

アクティブシミュレーションゲーム「ファーストクィーン」の続編がついにX68000に移植される。前作「ファーストクィーン」は多人数のキャラクターを扱うことができ，各々のキャラクターが細かいパラメータを持っているというRPGの要素をも含んだ，スグレもののゲームだった。「ファーストクィーンII」でも，前作で好評だったこのあたりのシステムには変わりない。ゲームの舞台は古代ローマとなっている。

内容に関してはPC-9801版からの完全移植とされているが，システム的に若干の変更はなされた。PC-9801版はIIになってマウスオペレーションが採用されたが，X68000版ではやっぱり前作同様キーボード，あるいはジョイスティックで操作することになる。もちろん，グラフィック周りやサウンドもX68000用に作り直されている。X68000ならではの出来を期待したいところ。PC-9801版ではセーブ数が1個だけだったのが，X68000版では3個までできるようになっているなどという細かい心配りもうれしい。

グラフィックに関してはもう一点追記すべきことがある。それは24kHzモードがあるということ。31kHzだと本来の画面の周りに枠がついたかたちで表示されるが，24kHzモードではスクリーンいっぱいにゲーム画面が表示される。

X68000用　5"2HD版　8,800円(税別)
呉ソフトウェア工房　☎048(646)0660

## スターウォーズと出たツイ，一騎討ち！

| | | |
|---|---|---|
| 1． | スターウォーズ | 1 |
| 2． | 出たな!! ツインビー | 3↑ |
| 3． | パロディウスだ！ | 6↑ |
| 4． | アクアレス | 4 |
| 5． | 生中継68 | 5 |
| 6． | パワーモンガー | 10↑ |
| 7． | ランス3 | ー初 |
| 8． | ジェノサイド2 | ー初 |
| 9． | キャメルトライ | 2↓ |
| 10． | イース | 9↓ |

と，まあ，このとおりの結果です。「スターウォーズ」の得票はますます増加中。2位との差をグングン広げています。「出たな!! ツインビー」も2位に順位を上げて必死に追っています が，スターウォーズをとらえることはできるのか。

また，今月は「パロディウスだ！」が3位にアップし，なんとコナミ勢が上位5作のうち3作までを占めています。もしも“GAME OF THE YEAR”にソフトハウス賞があったら，今年はコナミで決まりでしょうね。

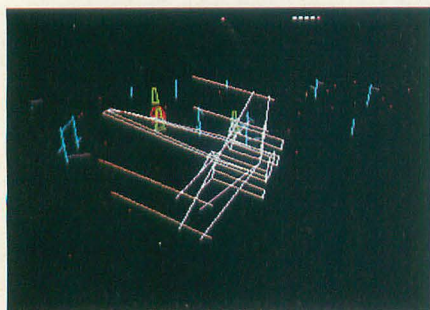
発売直後の反響はイマイチだったものの，しだいにハガキに名前を見るようになってきたのが「アクアレス」。最近の話題作にしてはめずらしいパターンですね。今月もがんばって4位をキープしています。ジワジワとプレイヤーの心をとらえているみたいですな。

「パワーモンガー」が先月の私の予想を裏切ってランクアップを果たしました。ややマニアックすぎるという声もありましたが，3Dで繰り広げられるリアルな世界に魅力を感じている人は決して少なくないようです。

今月の初登場は2本。ちょっとその2本の推薦理由を見てみましょう。

ランス3：ハードディスクに対応しているがんばりがいい。主人公の性格がなんともいえん。ゲームとしてのデキもよさそう。

ジェノサイド2：デモで兵士まで出てくるディテールの細かさがいい。ひさしぶりにわくわくするアクション。キャラクターのパターンが多い。格闘モノだから期待。

さて来月ですが，“GAME OF THE YEAR”集計のためTOP10はお休みさせていただきます。では再来月までごきげんよう。　(浦)

## レミングス

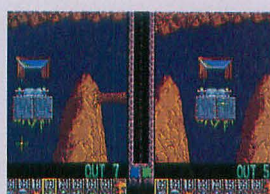
ぞろぞろと歩くレミングたちは，そのままだと海に飛び込んだり崖から飛び降りたりして死んでしまう。このレミングたちを救うのがプレイヤーの役目である。レミングに命令し（直接手は下せない），安全な道を作らせ，無事に彼らの家へと導かなければならない。状況は面を追うごとに苛酷になってくる。限られた数の命令を無駄なく使わねばならない面，制限時間が極端に短い面，そもそもどういう道を作っていいのかわからない面……これらが高度に組み合わさって，プレイヤーに挑戦してくる。周到な計画とマウスさばきが必要だ。 (A.T.)

X68000用　5″2HD版　9,800円（税別）
イマジニア　☎03(3343)8911

画面はAMIGA版です。

## スピンディジーII

“どんなゲームなの？”と聞かれると，“「マーブル・マッドネス」みたいなゲーム”と答えるしかないような気もするが，実はあまり似ていない。慣性の働くコマを操作して，さまざまな仕掛けをクリアしつつ，ゴールを目指す，というのは同じ。しかし，「マーブル・マッドネス」では仕掛けよりコースに重点を置いていたのに対し，この「スピンディジーII」ではむしろ仕掛けのほうに重点が置かれている。こちらのほうが頭を使うので長持ちするかな。

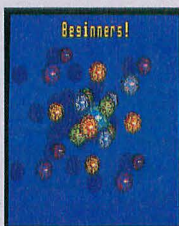
X68000用　5″2HD版　8,700円（税別）
アルシスソフトウェア　☎0956(22)3881

画面はAMIGA版です。

## ジェノサイド2

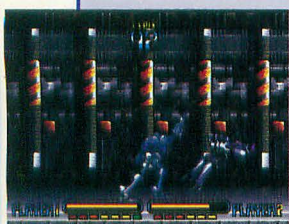
「ジェノサイド」でX68000の世界に飛び込み，次々と話題作を繰り出してきたズーム。そのズームがデビュー作の続編「ジェノサイド2」を発売した。

ゲームの中身は前回と同じく，ライトサーベルを手にトレーサーが暴れ回るという，アツいアクションゲームとなっている。自機や敵のアニメーションはさらに磨きがかかり，演出もなかなかのもの。

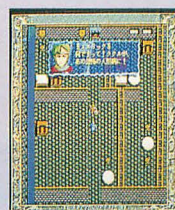
「ジェノサイド」は“スゴイけど，カタくてムズい，でも好き”というゲームだったが，今回もそのあたりは期待できそうだ。

X68000用　5″2HD版　8,800円（税別）
ズーム　☎011(613)0191

## ヘビーノヴァ

対戦型格闘ゲームというと，結構熱中してしまうジャンルであるが，この「ヘビーノヴァ」もそんなゲームのうちのひとつだ。もちろんひとりでも遊ぶことはできるが，このテのゲームはふたりで遊ぶのがスジというものだろう。各プレイヤーは，動力形式，スピード，攻撃法などにおいて特徴づけられた何種類かのヘビードールから自分のマシンを選ぶことができる。技もいろいろと用意されている。ジャブ，膝蹴り，ハイキック，飛び蹴り，回し蹴り，ズームパンチ，引き起こし，ミサイル，そして，多彩な投げ技。ちなみに返し技もあるぞ。

X68000用　5″2HD版　5,800円（税込）
ブラザー工業(TAKERU)　☎052(824)2493

## ブルトン・レイ シナリオ集VOL.3

システムソフトとコンプティークが主催した「ブルトン・レイ シナリオコンテストII」。このコンテストで最終選考まで残った20作品が「ブルトン・レイ シナリオ集VOL.3」として発売される。正統派の冒険活劇から，愛情溢れるお話まで，バラエティに富んだ作品が集まったようだ。「ブルトン・レイ」はシナリオエディタも発売されていて，遊ぶ楽しみと作る楽しみの2つの面を持っている。いろいろな人が作ったゲームを楽しませていただこう。

X68000版　5″2HD版2枚組　4,800円（税別）
システムソフト　☎092(752)5278

## ユニオン

ちょっと変わったパズルゲームが登場。牌と牌をくっつけると，縦2倍，あるいは横2倍に牌が大きくなる。大きくなった牌どうしをくっつけて，縦横ともに2倍という牌を作ることもできる。こうやっていって，すべての牌が大きくなると，その面はクリアということになる。少し地味な感じもするが，牌の模様や音楽も選べるし，なんといっても牌が大きくなるのが面白い。

X68000版　5″2HD版　7,800円（税別）
ポニーテールソフト　☎0722(85)2060

# 懐かしのアイドル再び!

Nishikawa Zenji
西川 善司

コナミがまたまたやってくれた。今度の出し物は「出たな!! ツインビー」。「パロディウスだ!」にも劣らない出来で，かなりクオリティの高い移植となっている。気合いが入ったサウンドやグラフィックには思わず唸るぞ。

「ツインビーありますか?」
「はい，CZ-217AS。7,800円です」
「うわーい。広告より安いやー」

……という人は，まずいないとは思うが，今回発売になったのは「**出たな!! ツインビー**」だ。発売元はコナミで9,800円(税別)。お間違いのないように。

## 出たよ!! ツインビー◆◆◆◆◆◆◆◆

では，まず外回りから。

今回の移植もかなり気合が入っているみたいで，「パロディウスだ!」並みにアーケード版に忠実な出来だ。アーケード版とは敵や雲の出現位置が微妙に違う，ということを誰かがいっていたが，縦画面のゲームを無理やり横画面に移植したんだから，微妙なバランスのくい違いやアレンジはやむをえないといえる(それにしても電波の「ドラスピ」以来，縦ディスプレイ対応のゲームに出合っていないが，またどこかがやってくれないかなぁ)。

また，そのためか画面が縦に潰れた感じを受ける。アーケード版を見慣れた人は，ステージクリアごとに入るアイキャッチのキャラクターやゲーム中の一部のキャラクターが妙に太ってしまっているのには結構びっくりするかも。ま，違和感を感じた人はディスプレイ垂直振幅ツマミをいじくるか，画面の45度斜め前からディスプレイを眺めよう。

さて，いちばん気になるゲームスピードも当然10MHzマシン用に調整されていて，X68000 XVI以外のユーザーも納得のいく動きを楽しめる。前作「パロディウスだ!」ではフグの拡大処理が実に苦しそうであった。私などは「ガンバレ」とかいって横からX68000をさすってしまったりしたが，「出たな!! ツインビー」ではああいった露骨なハードのウイークポイントの露出シーンもなく，コナミの技術力の美味しいところばかりを堪能できる。

## ナイスだ!! ツインビー◆◆◆◆◆◆◆

アーケード版のBGMがいいのは，すでに「(善)バビ」のコーナーで紹介ずみだが(ぜひCDのほうも聞いてみて)，X68000版もサウンドに気合が入っている。まず，うれしいのがMIDI対応だ。しかも，MT-32以外にSC-55 (CM-300/500も含む)にも対応している。これはもう手放しで評価したい。

実際にSC-55で演奏されているのを聞いてみて，私は「いい意味」で期待を裏切られた。「パロディウスだ!」のときのように，FM音源の足りないところをMIDIで補うような「原曲再生」でくると思ってたのに，まったく予想を外されたのだ。SC-55版では原曲のイメージを崩すことなく，しかも単にグレードアップバージョンに終わっていない，完璧な(?)編曲で演奏されていたのだ。たとえば，2面の「雲海を越えて」はCDのアレンジバージョンを意識したフュージョン調になっており，ところどころメロディのシンセギターがシンコペートしてるあたりはもう涙もの。エンディングの曲もハンドベルのリズムや，アコースティックギターのオブリガートに感動をそそられる(フレットノイズはよけいだけど)。

一方，MT-32版はSC-55とはひと味違った趣のアレンジで，コナミオリジナルの音色をふんだんに使って，LA音源のクセを感じさせない見事な出来栄えとなっている。どちらかといえばこちらは原曲に近いノリで安心して聴ける。

MIDIを使用しない内蔵音源のみの場合は，原曲からリズム以外のPCMパートをとっぱらったという，「パロディウスだ!」のときとだいたい同じような形態。原曲ではPCMパートで奏でられていたフィルの一部がちゃんとFM音源で鳴っていたりして，かなりがんばっている。

こうして聴いてみると，どの音源でもそれぞれ甲乙つけがたい素晴らしいゲームミュージックになっている。もちろん，ミュージックモードもあって，録音して楽しめるようにとのコナミの親心か，なんと「フェードアウト」機能まで搭載している。いたれりつくせりだよ，コリャ。

まあ原曲を重んじる人のなかには，今回のアレンジは結構ショックという人もいるかもしれないけれど，コナミがせっかくX68000のためだけにアレンジしてくれたんだから，いつまでも泣いてないでいいかげんにその顔を上げて，僕にとっておきの笑顔を見せておくれ。うひ。

ちなみにMIDI使用時は内蔵音源は効果音に徹する。このためMIDI使用時は曲も効果音も，とぎれなく演奏されるためたいへん気分がいい。もう時代はMIDIを求めているのか。はてさて……。

X68000用 5"2HD版2枚組 9,800円(税別)
コナミ ☎03(3264)5678

お城がきれい。ぶんしーん

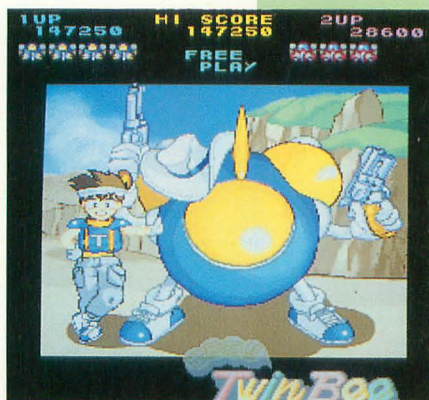
## やるぜ!! ツインビー◆◆◆◆◆◆◆◆

ストーリーはアーケード専門誌とかゲームセンターのインストラクションで知っていると思うので手短に。「惑星メルが惑星イーバの手で滅ぼされようとしていて，惑星メルの王女様メローラが助けを求めてきたので，ツインビーとウインビーが立ち上がった」以上。

「出たな!! ツインビー」は７年前の初代「ツインビー」の続編にあたり，冷静にゲームのみを見ればリメイク版といえるだろう。よって，ゲームシステムは基本的に７年前の初代「ツインビー」と同じだ。空中と地上の敵を破壊しつつ先へ進み，エリア最後のボスを倒して次のステージへ，というのが基本システムだ。もちろん２人同時プレイも可。1Pはツインビー，2Pはウインビーを操作することになる。地上攻撃用の腕が全部取れてしまったときに登場して腕を治療してくれる，あの「救急車」も出現するし，1P，2P手をつなげばワイドショットも出る。あのベル(鐘)パワーアップシステムももちろん健在。

初代「ツインビー」を知らない人のために，この「ベルパワーアップ」を簡単に説明しよう。まず雲に隠れたベル(鐘)をショットで雲の外に追い出し，さらに撃ち続けるとベルの色が変化していき，これを取るとその色ごとに定められた機能が自機に装備されるというもの。「出たな!! ツインビー」では新たに２色のベルが初代のシステムに追加された。

黒のスピードダウンと，紫の「しっぽバリア」がそれ。前者は読んで字のごとし。後者は自機の分身が現れ，それらがシールドになって敵や敵弾を防いでくれるというもの。加えて，以下の２点は頭に叩き込んでからプレイしてもらいたい。

**1) ベルは５発撃つごとに色が白→青→緑→赤→紫→黒と変わっていき（各色の間は通常色である黄色），撃ち続けると最後には蜂（破壊不可の敵）になってしまう**

**2) 「バリア」（赤ベル）と「しっぽバリア」（紫ベル）は基本的に同時に取れない。ただし両方つけていないときに赤と紫を出し，この両方のベルを取れば，両方装着できる**

楽しいアイキャッチシーン

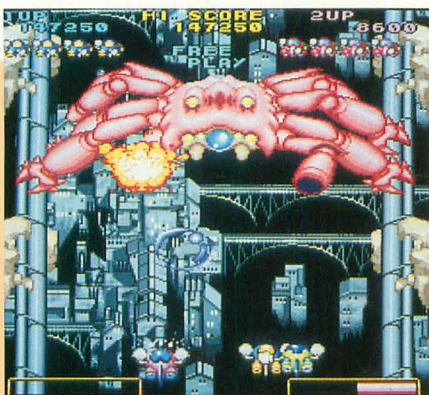
救急車登場。ありがたき幸せ。

2)は，つまり「バリア」を装備しているときは紫ベルが，「しっぽバリア」を装備しているときは赤ベルが出ないということ。これって重要。

それにしてもこのベルパワーアップシステムはツインビーのゲーム性をいっそう高くしているよね。適度にベルパワーを取らないといけない，けれどそのことにばかり集中していると敵にやられやすくなる……。なんとも心憎い，うまい方法を考えたものだ。

ところで，色と機能は「パロディウスだ！」のそれとはいっさい無関係。緑を取っても巨大化したりしないからね。

もうひとつパワーアップチャンスがある。地上物を破壊したときに出るフルーツ。これは単なる得点になるだけ。しかし，たまにパワーアップアイテムが出ることもあり，これを取ることによっても自機はパワーアップできるのだ。

★マークは画面内の敵を全滅させ，チビベルマークは自機が３ウェイショットになる。そしてグインビーマーク。これを取ると「グインビー」というキャラクターが浮上し，ふらふらと画面上部へと浮遊する。これを捕まえるとプレイヤーは「グインビー」と合体し，ワイドな「グインビーショット」が撃てるようになる。さらに２人プレイのときは，グインビーと合体しているほうにもう片方が体当たりすることで，「グインビーアタック」を仕掛けることができる。つまり，グインビーがひとりで動き，敵を追いかけてやっつけてくれるのだ。ボスシーンや敵が溜まったときにやると，見た目も楽しいしすごく助かっちゃう。

多彩なパワーアップができるのはわかってもらえたと思うが，実は「出たな!! ツインビー」の自機は，初期状態から初代「ツインビー」にはなかった強力な新兵器を装備している。なんとボタンを押しっ放しにすることでエネルギーを溜めることができ，その溜めたエネルギーを一気に放射することができるのだ。つまり，波動砲というやつ。エネルギーマックスの波動砲は大変強力で，発射した瞬間には自機の真後ろや真横にいる敵にもダメージを与えられる（俗にいうバックファイヤ）。

さらにさらに，２人プレイ時には自機どうしを縦に衝突させることによって「ガッチンコ」攻撃というものができる。これは衝突の際に飛び出す「ガッチンコ」の５文字を放射状に飛ばして敵を粉砕する，愉快かつ強力なショットなのだ。

これが「ガッチンコ」ショットだ！

カニのアワに当たると動きが鈍くなる

## はじめるぞ!! ツインビー◆◆◆◆◆◆

**とにかく，**

「出たな!! ツインビー」は２人でプレイしたい。気の合う友達同士なら，なかよくやりたいもの。でも，

「その赤ベル，お前にやるよ」

「悪いよ，お前が取れよ」

「……ポッ♥」

なんてことになっても，当局は一切関与し

左端にいればカミナリに当たらない

流れ落ちる水が美しい。光るのは水晶か？

ないからそのつもりで。イヤなやつとならプレイ中，リアルタイムにF1，F2キーを押して解像度を切り替えてやるくらいのイヤガラセは必要。恋人どうしなら……，ゲームなんかしてる場合か，すべきことがあるだろ，ほかに。

……前置きが長くなった。

さて，ゲームを開始すると「いかにも」お姫様という感じの美少女が救いのメッセージを投げかける。なんとなくビデオシステムの「ラビオレプス」に似ている（といっても誰もわからないか）。

「私は惑星メルのメローラ」だって。ほとんど「私はテレザートのテレサ」のノリだ。

ステージ1は風の渓。自機を確認したらボム（地上攻撃）ボタンを連射にセットしちゃえ。ショット（対空攻撃）ボタンは溜め撃ち（波動砲）ができなくなると困るから連射は解除。エッ，キーボードの人……？ キーボードの人はこの先，辛いぞ。指立て伏せ100回くらいの準備運動がないと，ステージ3あたりで指が吊るかもよ。お金が余っている人はいますぐジョイスティックを買いにいこう，技術のある人は作っちゃえ。技術もお金もない人はマミーやダディーに「おねだり」するか，大道芸でもやって稼いででも買ってほしい。

顔のついたお茶目な雲の大群がやってきたら，ベルをショットで追い出して，さあ，パワーアップ。ステージ3くらいまでなら，結構冷静にベルパワーをセレクトできるだろう。逆にそれ以降は，生き延びることを優先に考え，黄ベルで得点しよう（最高時1万点は美味しいよ）。まあ，でもスピード1速に「分身」は最低でも身につけていたいな。

標準で波動砲があるんだし，ツインショットは“取れたらラッキー”程度に考えておこう。地上攻撃をキチンとしていれば，どうせそのうちベルマークの「3ウェイ」が手に入るだろうしね。バリアは前述のとおり2種類あるわけだが，私は扱いやすい赤ベル「バリア」をお勧めする。

平原を過ぎて崖が見えるあたりで「王」マークのついた雑魚が現れるが，これってもしかして，あの電話をかけてくるという「ウンモ星人」のマークでは？　コナミの社員には，相当なUFOマニアがいるみたいだな。

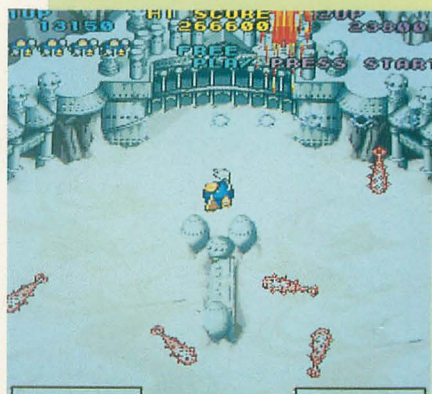
崖から鉱山地帯へ。BGMがフェードアウトしていき，デンジャラスなボスのテーマとともに，馬鹿でかいカニが登場。それにしても動きがすごい。ステージ1からいきなり見せてくれる。でも，ひとつ注意。波動砲を撃つとスプライトオーバーかなにかの原因で，敵の弾が消えることがあるのだ。あらかじめ敵弾の弾道を予測していないと消えてる（？）敵弾に当たったりして，わけもわからず死んでしまうことがある。あまりシビアな弾かわしはしないで，余裕をもって避けよう。こういった現象は2P時にちょくちょく起こるが，やはりハードの限界なのだろうか。

## きれいだ!! ツインビー◆◆◆◆◆◆◆

ステージクリアをするとアイキャッチのグラフィックが表示される。通常こういったことをすると，それまでに高められた世界観をブチ壊しかねないのだが，ゲーム本編のコミカルな雰囲気と統一がなされているために全然違和感がない。逆に次のステージへの期待が膨らむくらい。失敗している例といえば，同じコナミから出ているアーケードゲーム「ゼクセクス」とか。ゲームは面白いのにね，なにか変だよ，あれは(知らない人は一度やってみよう)。

ステージ2は雲が流れる美しい森の上空からスタートする。雲間から見える小さい木々や岩肌が高度とスピード感を演出。ズームの「ファランクス」にしろ，この「出たな!!　ツインビー」にしろ，最近のゲームは実に背景が美しい。もはやCGアートの世界といえる。

さて背景に見惚れていないで，1面で装

ベルに気を取られてやられないように

なつかしの回転「こん棒」

### いたれりつくせーり

ゲーム中にESCを押してF3キーを押すと，疑似縦画面（384×256ドットモード）になる。CU-21HD（21インチCRT）でないと迫力がないけれど，コナミの親心がなんとなくうれしい。

これだとかなりアーケードに近い縦横比になり，ツインビーちゃんの顔も斜めから見なくても細く，やせてる。また，2Mバイト以上のメインメモリがある人は，起動時からHELPキーを押していれば，なんとゲーム中ディスクアクセス一切なしのオンメモリになっちゃう。

ディスク1には「SD_DRV.BAT」という，特設ミュージックモードや愉快な開発後記ドキュメントが入っているし，さらにはハードディスクへのインストール用バッチファイルまでも。よっコナミ，日本一!!　またなにか作ってね。

（欲深い善）

| 総合評価 | 0　　5　　10 |
|---|---|
| ゲーム性 | ★★★★★★★★★ |
| グラフィック | ★★★★★★★★★★ |
| サウンド | ★★★★★★★★★★ |
| 2人プレイ | ★★★★★★★★★ |
| お買い得度 | ★★★★★★★★★ |
| 熱中度 | ★★★★★★★★★ |

備を失った人は序盤の雲群でパワーを再装備しよう。雲からペンギン（なぜか対地攻撃が有効）が攻撃してくるがこれはだいたいの場所を覚えて先撃ちしよう。見逃すと後ろから狙い撃ちされることもあるし，甘く見てるとひどい目に合う。サインカーブを描いて迫ってくる緑のソラマメは，マックスの波動砲で編隊を丸ごと破壊しよう。変に残すと画面外から衝突されたりしてやっかいだよ。

ところで，波動砲のエネルギーゲージは画面下にあるが，いちいちゲージを見ている余裕はない。そこでエネルギーがどれくらい溜まったのかは，エネルギー充塡中の効果音「ウイ～ン」で判断する。「ウイ～ン」の「ン」までが鳴り終わったときにちょうどマックスになっているはずだ（なんだかマニアックなお話）。

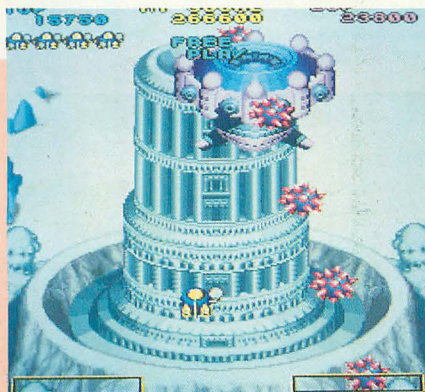
画面に居座るイカリングのあとは，凶悪な折り鶴が登場する。間違っても出現位置にいないように。

台風の目のあとはピンクの象の特攻隊，マックスの波動砲をタイミングよく発射していればまず大丈夫。ボスは巨大戦艦だ。回転を始めたプロペラを抜けるときには，行きを左から，帰りは右からがベスト。帰りは画面左下のすみっこにいるだけで大丈夫だ（安全地帯）。

ステージ3，天空の要塞面。雲と要塞が絡み合う多重スクロールの背景は見応えがある。敵の弾は見にくいけど。

序盤はまだ余裕を持って復活に専念できる。画面の左右両端からのタコ編隊は，片側だけでも波動砲で一掃しておかないと画面に溜まってたいへん。

グラディウスでおなじみのワープ部隊や後ろから迫るずるーい蚊の大群は場所を覚えないとダメ。ここはまさに初心者トラップのシーンといえる。

ボスやその手前のワープ部隊で死んだりするとステージ4の序盤がつらい。ステー

塔からボスが登場

にっくきライバルツインビー

ジ4の序盤は雑魚のクラゲや緑のボールがじゃまをしてなかなか復活ができないからね。

そのステージ4はキノコや苔の世界。雑魚キャラも耐久力がついてきて通常弾1発じゃ死なないのも出てくるし，かなり難しい。特に空飛ぶ三葉虫は確実に自機の位置を狙ってくるので要チェックだ。ボスは稲妻を3発放ったら充電で休む，画面左端には稲妻は来ない，ということを覚えておこう。

ステージ5は水と水晶の世界。ステージ4のボスで死んだ人は，始まってすぐにある雲でなんとか基本装備を整えよう。もうこのへんまで来ると分身よりも前にスピードかバリアを先に装備したい。

初代「ツインビー」で懐かしい包丁ザコは前作同様のいやらしさ。ステージ3のタコ同様に画面の左右から登場するので，同じく片側だけでも波動砲で一掃しておきたい。ボスは最近のシューティング定番の多関節キャラ。破壊した関節は当たっても平気だから，無理して弱点の頭ばかりを狙わず，移動範囲を有効に使って落ち着いて倒そう。青いホーミングは波動砲で消せる。

ステージ6は「ドラスピ」の4面を思わせる砂漠地帯からスタート。ここまで来ると，もうどこでも平均的に難しいわけだが，

背景の雲はラスタスクロール

2周目はパレットが変わる

チェックしておきたいのは終盤の「こん棒」が回転しているシーン。同時に，ステージ3で登場した蚊の編隊が後ろから来ることだけは忘れるな。「こん棒」を避けることに集中しているととんでもないことになる。この「こん棒」にしろ包丁や牛乳瓶にしろ，初代「ツインビー」のキャラクターを織り交ぜて出現させる演出，昔ながらのゲームファンにはこたえられませんな。

残すは最終ステージだが，ここからは各自，自力で切り抜けてくれ。最後のボスは私も完璧な攻略法が見えない（なはは）。腕達者の君のレポートを待ってるよ。よろしく，たのむ。

### 遊び倒せ!! ツインビー◆◆◆◆◆◆◆

2周目は各面，背景のパレット（色）が変えてあったりして，見た目が新鮮。ただ難しいだけの2周目じゃないっていうのがうれしいよね（って実際やってみると，ただ難しいだけなんだけれど）。

コンティニュー回数が無限なので，誰でも最後まで行けるだろうけど，1万円近くするソフトを一度最後まで遊んだだけで「ポイッ」ってのはもったいない。

通常プレイに飽きた人に私が提唱するのは，1面トライアルとノーコンティニュートライアル。

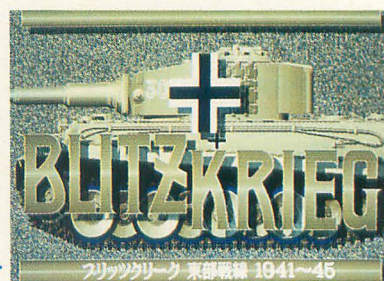
前者はステージ1で何点取れるか，後者はコンティニューなしで何点取れるかを競うもの。インタラプトスイッチでタイトルデモに戻れるのを利用して，何度も何度も練習しよう。

友達を集めて新年「出たツイ」大会も悪くないし，ひとり寒い部屋でかじかんだ手をほぐすために，寒中「出たツイ」も捨てがたい。ちなみに私はノーマルランク，3機設定で前者が235,500点，後者が889,500点が最高だった。

それじゃ皆さん今年もよろしく。

# 電撃戦は雷帝とともに

Kaneko Shunichi
金子 俊一

**見た目は「大戦略」シリーズと似たようなものなんだけど，中身はちょっとマニアック。といっても，難しいとか，複雑だとかいうのではなく，戦車での戦いをクローズアップしたシミュレーションゲームなのだ。**

ブリッツクリーク（電撃戦）とは，第2次世界大戦においてナチスドイツ軍とソビエト軍が戦ったときの作戦概念であり，東部戦線では「バルバロッサ作戦」，「スターリングラード攻防戦」などが有名である。「ブリッツクリーク」はこれらの史実をもとにしたシミュレーションウォーゲームである。

## 島国ニッポン◆◆◆◆◆◆◆◆◆◆◆◆◆

プレイヤーはドイツ軍側と決まっている。必然的にコンピュータはソビエト軍である。ただしドイツ軍にはプレイヤーと別行動をとる部隊もいて，それはコンピュータが受け持つ。それを友軍ドイツ軍と呼ぶそうな。つまり，人間＋コンピュータ対コンピュータの戦いになる。

第2次世界大戦中の兵器といえば，日本人なら戦艦大和や三菱零式艦上戦闘機（ゼロ戦）などを真っ先に思い浮かべるだろう。別に空母飛龍や，B52と竹槍でもかまわないのだが，やはりそれらは島国ならではのことである。このゲームの舞台であるヨーロッパ大陸では，戦車が主な兵器であった。

このゲームでも戦車戦がメインであり航空兵器は支援にしか使われない。だからといって，「なんだ，戦車だけか」などといってもらっては困る。実に細かい仕様になっているのだ。ドイツ軍の4号戦車だけでも，C型，F1型，G型，H型があり，それぞれの特徴を押さえているようだ。もちろん，4号駆逐戦車などもあり，戦車ファンにはこたえられないだろう。

このゲームでは，それら戦車などの兵器を5台集めて小隊と呼ぶ。小隊を5隊集めると中隊という単位として数え，基本的に中隊ひとつに対して指令を出す。よってひとつのマス目に中隊ひとつである。

実際にゲームを始めると小隊に満たない小隊や，中隊に満たない中隊がぞくぞく出てくる。ようするに破壊されてしまったぶんは空きができるわけだ。合流というコマンドで，足りない中隊どうしを合わせて，ちゃんとした中隊にすることもできる。

補給部隊がないのも面白い。補給は自軍の街からの距離に応じて受けることができる。だから，立ち往生してしまった兵器はもう二度と動くことはないのである。

ちょっと目新しいのは，いらなくなった中隊を破棄することができるようになったこと。ガンダムでいえば旧ザクのような「実戦には使えなくなった兵器」などは捨ててしまえる。まあ，さほど使うコマンドとは思えないが。

## 勝利の女神に投げキッス◆◆◆◆◆◆◆

大戦当時のドイツ軍は戦場を転戦していた。それらの各戦場を舞台にして話を進めていくのがシナリオ制である。

ひとつのシナリオにはマップと勝利都市と制限ターンが設定される。制限ターンが終わったときに勝利都市を確保していると，そのシナリオを制覇したことになる。もし制限ターンより前に勝利都市を占領すると，その都市を防衛する必要があるということだ。ちなみに最初から勝利都市がドイツ側の場合もあり，それは都市防衛が任務になるといえる。

プレイヤーは数十個中隊を指揮下に置く，1個師団の師団長という設定である。まあ最前線を任せられるようじゃ，たたき上げの軍人なのだろう。与えられた兵器で，上からの作戦を忠実に実行していくしかないようだ。

シナリオ制で面白いのは，史実に基づいているため，使える兵器が完全に決められていること。選択の余地はなく，このシナリオならこの戦車とこの兵隊，というふうに決まっている。もちろん，次のシナリオで「新たに開発された兵器が投入される」こともある。

この時期（1940年頃）の戦車は開発途上中であり，最新型の戦車が開発されると，いままでの戦車ではまったく太刀打ちできないこともあったようだ。それはまるで車田正美のマンガの必殺技のように，理不尽なまでに強かったりするのだ。テリオスとはいわない，ブーメランフックで人を倒していた頃が懐かしい。わからなければブロンズ聖戦士とゴールド聖戦士の違いだと思いねえ。

X68000用　5"2HD版2枚組　9,800円（税別）
システムソフト　☎092(752)5278

スタート画面も雰囲気たっぷり

マップの全体図を見る

このゲームでは，最初の頃はソビエト軍のT34/76がむちゃくちゃ強い。当時のドイツ軍最強である4号戦車C型をもってしても傷ひとつつかないようだ。こいつは困った。次のシナリオでやっと対抗できる戦車を持つと，ソビエト軍はさらに最新鋭のKV-Iを投入してきた。クワバラクワバラ。歯が立たない。

こういったシナリオ制ならではの苦労もあること以外は，ごく普通のシミュレーションウォーゲームのようだ。

移動できる範囲を見ながらよく考える

部隊が未行動か行動ずみかもすぐわかる

システムはHEX戦である。大戦略シリーズでは当たり前とされている，テニスコートの金網のようなあれだ。ただし，HEX戦にアンダースコートは見えない。

表示画面は2種類あり，19×19と38×38である。19モードでは兵器の戦闘力やシルエットなどがわかる。38モードでは兵器の分類しかわからないが，大局を把握しやすい。PC-9801版が15×15か30×30だったので，X68000の画面モードを有効に使っているということがわかる。

ちょいと問題なのが，表示画面の移動。画面の右上のほうに見えている全体図で白ワクを移動させるのだが，マウスの移動量が大きく，面倒臭い。HEXのいちばん外側をクリックしても画面は動くが，このへんは一発，「銀河英雄伝説」あたりを見習ってほしかった。よくできたシステムを真似することは決して恥ではないと思う。中身で勝負してもらえば，それで問題ない。最初のうちは右ボタンをクリックしたまま動かそうとしてたっけ。

そのほかのところも操作に関してはあまりほめられない。慣れればどうにでもなるが，そういう問題ではないはずだよね。

## マニュアルは要しおり◆◆◆◆◆◆◆◆

シミュレーションゲームはマニュアルが厚いというイメージがある。このゲームでもそれは確実に当てはまる。マニュアルは67ページ＋付録17ページ，導入ガイドは12ページ，ヒストリカルノートは66ページ。厚い。マニュアルを読むのに「しおり」を使ったのは初めてである。

いままでも厚いマニュアルは読んできたが，しおりはいらなかった。今回はどうもマニュアルのほうにしおりを使わせる原因がありそうだ。

それはマニュアルが初心者向きにできているからのような気がしてならない。

別に上級者向きに作れといってるのではない。このゲームが中級者向きなのだから，中級者向きにマニュアルを作ればいいのだ。さらに初心者用のガイドなり，用語解説を別冊でつければ問題はないのだ。

難しい本は読んでて眠くなるものだが，やさしすぎる本も眠くなるのだ。

マニュアルをわかりやすくしようという試みをやっていることはわかるが，余計な囲みを入れたりして，中途半端に編集に凝ったぶんだけわかりにくくなっている。

「過ぎたるはなほ及ばざるがごとし」

この名言を送ろう。

## コンピュータはO型？◆◆◆◆◆◆◆◆

最後に，肝心なコンピュータの戦術レベルについて触れておこう。パッケージにあるような難易度4（当社比）というわりにはちょっとふがいない。多少の改善はあるものの，結局いつもの力押しパターンである。

ただし，甘く見てはいけないところはある。実例を紹介しよう。

私が守っていた都市は東と南の方向にソビエトの街があり，北と西には街らしきものは何もなかった。よってほとんどの中隊を東と南だけに割り振っていた。事実，東からの攻撃はすさまじく，何個小隊が戦場に散ったかは数えきれないほどだった。ソビエト軍の攻撃がやっと落ち着いてきたころ，突如として友軍ドイツ軍が北に向かい出したのである。まさか？　と思いつつ態勢を整えていると，数ターン目にして友軍が敵と接触したのである。あわてて少ない戦力を割いて，北に向かわせたのはいうまでもない。

このことで2つのことがわかった。

ひとつは，こちらの配置図を把握し，弱点を狙ってくる賢さがあること。

もうひとつは，コンピュータは友軍も座標を全部知っていること。これはちょっとズルいね。

結末はというと，北に回ったソビエト軍は燃料切れで立ち往生していた(笑)。もっと愚かなのは友軍。止まっているソビエト軍最新鋭戦車に向かって攻撃を仕掛け，ボロクソにやられてしまったのだ。つまり，コンピュータは弱点を突くことはできても，燃料とか細かいことを考えない猪突猛進タイプで，人間でいうならO型の人にありがちなタイプであることがわかる。もう少し緻密に思考してほしいね。

村を占領した。シブメのグラフィックだ

### ヒットラーはアウトバーンを作った。さて，システムソフトは？

そろそろHEX戦以外の方法も検討してほしいところ。HEX戦というのは，あくまでもボードゲーム用の方法であって，コンピュータにはコンピュータなりのシミュレーションゲームがあってもいいはずである。このゲームではHEX表示のオン/オフがあるが，根本的な解決になっていない。画面がきれいならいいという問題ではないはずだ。そのほかにも，細かい点がいろいろと気になるが，これらが改良されればかなり面白くなるだろう。

とにかく，単にPC-9801からのベタ移植ではないところには可能性を感じるのである。

| 総合評価 | 0 — 5 — 10 |
|---|---|
| システム | ★★★★★★★ |
| ミュージック | ★★★★★★★ |
| サウンドエフェクト | ★★★★★★★★ |
| マニュアル | ★★★★ |
| 発展性 | ★★★★★★★★★ |
| 戦車フェチ | ★★★★★★★★★ |

# 蒼き鮫，天地を制す

Yaegaki Nachi
八重垣 那智

若干遅れはしたけれど，ついに，「飛翔鮫」が発売された。降り注ぐ弾を避けつつ，地上と空中の両方の敵を破壊し，危なくなったら「ボンバー」を使って切り抜ける。しかし，多用はするな。「ボンバー」は最後の武器だ。

1987年の春，ちょうど世の中にX68000が登場した頃にゲームセンターを覗くと，そこには鮫がいた。レトロな感覚の中で熾烈な戦いが繰り広げられる熱いシューティングゲームが，挑戦者を待っていたのである。そして常に敗者となる運命を知りつつも，私はその鮫に100円玉という餌を与え続けていた。その当時を思い出すならば，この「飛翔鮫」というゲームを抜きに語ることはできないくらい，印象深い思い出のゲームなのである。

## 伝説の鮫，降臨す

ゲームセンターの秀作ゲームは，実物をプレイしていなくても各種の家庭用ゲーム機に移植されているので，そちらのほうをプレイすれば若干の相違こそあれ，同じゲームの楽しみを多くの人が味わうことができる。もちろんX68000に移植されているゲームも少なくない。しかし，この「飛翔鮫」の国内での移植は，(私の記憶に間違いがなければ)今回のX68000版が初めての試みであるはずである。そういった点で，このゲームの知名度はやや低いかもしれないので，少々詳しく説明してみよう。

ゲームの操作からいえば，レバーかテンキーで移動して，ノーマルショット，および，緊急回避と強力な攻撃を兼ねたボンバーをボタンで発射する，オーソドックスなタイプということができる。が，実は本末転倒である。こういったタイプがメジャーになったはしりがオリジナル版の「飛翔鮫」であり，すべてがここから始まったといっても過言ではないからである。地上物と空中物をどの武器でも同時に攻撃でき，強力なパワーアップをも可能になったのが，まさにこの「飛翔鮫」なのである。そういった意味では，このゲームにとても単純で純粋なシステムの美しさを見いだすことも不可能ではない。

その自機のパワーアップであるが，当時としてはめずらしくスピードアップを廃したかたちになっているのが特徴である。アイテムとしては基本的に2種類しかなく，敵の赤い編隊を全滅させると現れるSアイテムで通常ショットを強化し(X68000版では4回取るとフルパワーになる)，特定の地上物を破壊すればBアイテムが登場し，これでボンバーを補給することができる。注意したいのはBアイテムは画面に1個しか表示されないので，アイテムを持った敵を続けて倒すと最初の1個しか出現しないことである。またステージをクリアすると3個に戻されてしまうので，ピンチには惜しまず使ったほうがよい。爆発すると，敵の弾を相殺できることも知っておくべきである。

面の構成としては，一定エリア数の敵の攻撃から生き延びて，飛行場までたどりつけばステージクリアとなる。特に面の最後にボスといった性格のものは登場しないが，デカキャラ的なものは随所に数多く登場するようになっている。全部でステージ5まであり，そこをクリアするとステージ2～5を繰り返しプレイする仕組みになっている。もちろん，敵の攻撃は激しくなっていくので，自分の限界まで挑戦することも可能である。

## 牙を剝き，天空を泳げ

と，まあオリジナル版と同じところは以上のようであるが，このX68000版「飛翔鮫」には残念なことにいくつかの移植に際しての変更点がある。些細なものから，目立ったものまでいくつかあるので，ここでまとめてみようと思う。

まず自機の速度がやや速く感じることである。これは画面が横になったせいであろうが，同様に自分のショットが画面外に出ていくのも早くなっており，オリジナル版よりも動きがスムーズに感じたり，連射の反応がよかったりするのでいい面もある。

パワーアップについてはかなり変わっており，最強の状態がオリジナルとは異なっている。気になるのはパワーアップアイテムを持つ敵編隊の登場パターンで，本来なら赤い編隊と黄色い編隊がほぼ交互といった感じで登場するのに，このX68000版ではフルパワーになるまで，必ず赤い編隊が登

X68000用 5″2HD版 8,800円(税別)
金子製作所 ☎03(5261)2147

簡単に出現する1UPの白い編隊

1ステージクリアして飛行場に着陸

場する。また，よほど運がよくないと見ることすらできない1upアイテムも，決まった法則で出現してしまうのは，どうかという気がする。

ゲーム中では，オリジナル版にあった破壊可能な地上構造物が破壊不能に変更されていることが挙げられる。本来なら戦車が出てくる建物などは，天井を破壊することができ，中にいる戦車が出てくる前に先制攻撃ができたのだが，そのフィーチャーはカットされている。まあ，これはこれでそんなに気にはならない性質のものである。

ボンバーは素早く取れ

巨大爆撃機登場。ここをなんとか乗り切れば……

これらの相違点はあらかじめ知っておきさえすれば，オリジナル版とのギャップに悩まされたりすることも少なくなるだろう。そもそも横画面になっていることで発生する，感覚の違いを考えれば，それほど問題ではないともいえるからである。

## 屠れ,地の鉄屑を ◆◆◆◆◆◆◆◆◆◆◆

それでは各ステージを紹介していこう。攻略というより，気をつける目安だが。

**●ステージ1**

最初の状態は弱いので，慣れないうちは深追いを避けよう。序盤の湿地帯での左側の木の下にいる戦車は要注意。大型戦車は時間をかけると，横から戦車が出てきて逃げ場がなくなるので注意。

**●ステージ2**

空母の上の大型機はそんなに恐れずにさっさと撃墜するべし。大型戦車のあとにあるボンバーを拾いにいくときは後方に注意すること。

危険なときは迷わずボンバー

ショットを強化すると弾が広がる

**●ステージ3**

なにがなんでも小型艇に尽きる。特に後ろから出てくるのは場所を覚えて全滅させよう。大型戦艦の上から復活するときはボンバーを惜しまず，効果的に使うこと。大型戦艦は再び下から登場するので注意。

**●ステージ4**

列車砲が曲者。慣れないうちはボンバーを連続で射ち込んで，確実に仕留めたほうがいい。機関車の貨車にはボンバーが入っているので，画面上に1個しか表示されないことに注意して補給すること。

**●ステージ5**

最終面だけあって結構シビア。港の戦艦2隻に始まり，後ろからの大型戦艦，超巨大爆撃機とヤマ場のメジロ押しである。しかも最後に，戦闘機の大編隊と戦車の大群が待ちかまえている。あまり簡単に後ろに下がりすぎると，逆にピンチになるので大胆に攻めよう。

## 戦いの果てで眠れ ◆◆◆◆◆◆◆◆◆◆

オリジナル版の「飛翔鮫」では，画面を振り子のように往復して，まんべんなく攻撃して敵の弾を誘導する，通称「切り返し」という基本テクニックが使えたのだが，このX68000版では，残念なことにあまりこれは通用しない。それというのも敵の弾が本来ならば，きわめて正確に自機を狙ってくるハズが，実際にはややズレたところにも飛んできてしまうために，単純に動いているだけでは逆に追い詰められたりすることがあるからなのである。

これは考えてみると非常に大きな違いで，シューティングの感覚そのものにズレを生じさせてしまうはずである。だがしかし，最初に挙げた違いのうちのひとつである自機の操作感覚の違いとうまい具合に調和していて，ゲーム性をそれほど損ねているといいきることには，ためらいがある。簡単にいってしまえば，画面を横に変更したことに対して，ゲームシステムのバランスを取り直したというように考えてみるといいだろう。現に私は，これはこれなりに「飛翔鮫」という名前を除いて考えれば，それなりに遊べるものと感じている。

しかし，移植という観点から見れば，これはあまり褒められたことではない。完全移植を希望する声は，オリジナル版が存在するかぎりなくなることはないからである。画面の幅が狭くなってもかまわない，あの思い出の「飛翔鮫」が自分のX68000で，まったく同じ感覚で遊べることを期待していた私は，複雑な思いでこのゲームをプレイしたのである。しかし，これはやはり「飛翔鮫」であり，数々の思い出が懐かしく浮かんでは消えていったのであった。

### 空にきらめく戦いの翼

金子製作所はこれからも東亜プランのシューティングゲームシリーズを移植していくらしいが，贅沢をいわせてもらうと今回の「飛翔鮫」よりもさらに何倍も優れた作品を希望したい。

昔から東亜プランの作ったゲームに（会社はタイトーだと思っていたが）触れ続けてきたゲームプレイヤーとして，それぞれの作品は当時の思い出が一緒に詰まっているからである。

残念なことに女の子と一緒にいい思いをしたという，甘い思い出のようなものがないことは，非常に残念だが，そのゲームを見ると失恋の思い出で涙が出るとかいうのよりはいいか，うんうん。

**総合評価**

| 総合評価 (0〜10) | |
|---|---|
| ゲーム性 | ★★★★★★★★ |
| 技術 | ★★★★★★★ |
| 移植度 | ★★★★★★★ |
| サウンド | ★★★★★★★★ |
| ボンバー射って死亡 | ★★★★★★★★★ |

# かわいきなかにも痛さあり

この「NIKO²」はいままでのウルフ・チームのゲームとは，かなり違った路線になっている。ジャンルも何に分類すればいいのかがよくわからないけど，全体的に“かわいくて，楽しい”というコンセプトで作られているようだ。

Komura Satoshi
古村 聡

いまの世の中，みんな痛い話は好まないけど，“かわいー”ものが“おいた”をするのは，ますます“かわいー”となって，歓迎を受けてしまう。

「徹夜明けでネボケてて，カミソリで思いっきり歯を磨いちまったい」などと，マトモに痛ーい話をすれば，10人中10人が“あんたあっち行ってよ”とイヤーな顔をするのは間違いないんだけどね。

さて，今回このページでご紹介申し上げますのは，まさしく，

かわいい＋いたい＝ますますかわいい路線を狙った対戦アクションゲーム，「NIKO²」(ほーらゲームのタイトルまでその路線だ) なのであります。

## どうだかわいいだろう ◆◆◆◆◆◆◆◆

このゲーム，あえていうなら，2匹のウサギ対抗タマ入れ合戦かな。

ゲームをスタートすると，スターターと呼ばれる物体から玉がバラバラバラバラっと出てきて，そいつをキャラクターの体で弾いては自分のゴールに入れる。制限時間が過ぎた時点，あるいは300個の玉すべてが画面上からなくなった時点で得点の多かったほうが勝ち，というルールなんだけど，単純にそれだけではない。

巨大化すると妨害しやすい

X68000用　5″2HD版　7,800円(税別)
ウルフ・チーム　☎03(5394)5565

この自分が操る，ぬいぐるみのよーな，ウサギのよーなキャラはタマを弾くだけではなく，まず，

**壁を作ることができる！**

ようするに敵のゴールに入りそうなタマを跳ね返すための壁を作れる。そして，

自分が不利になると大きくなる！

点差が開くと，だんだん負けているキャラの大きさがでかくなる。こうなると玉を弾くのも楽になるし，作れる壁も大きくなるからジャマしやすくなるぞ。

そして，このゲームでの最大の特徴はなんといっても，

**なぐる！**

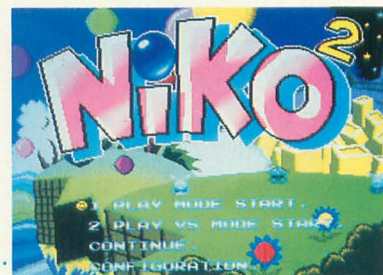
であるのだな。相手をなぐると，“うるる～ぅ”の声とともに一定時間気絶する。

## 思うにルールがアツい ◆◆◆◆◆◆◆◆

このゲームは“うるる～ぅ”“にこにこ”“ぼよよん”(にこにこ，ぼよよんはそれぞれのゴールに玉が入った音) という怪音が飛び回るゲームだ。いや，というより，

**に，に，にこにこ，ぼ，うるる～，ぼ，ぼ，にこにこぅ～**

と，実にやかましいゲーム，といったほうがぴったりくるかな。どうだ，“かわいー”気がしてきただろう。

キャラクターの感じといい，ゲームのネーミングといい，ひとつ間違えば，まるで同人ゲームという感じのものを，これだけ

この背景はさすがにボールが……

アツくなるゲームにできるというのはやっぱりプロのなせるわざなのだろうか。

体のどこに玉が当たるかでどこにボールが跳ね返るかが決まるので，うまくいけば鼻先に当ててことごとくボールをゴールに入れる！　なんて荒ワザもできるが，半面，跳ね返し方をひとつ間違えるとボールは相手のゴールに入ってしまう。間違えて相手ゴールに玉を入れてしまったときはくやしいのなんの。コンピュータはきっちり思ったところに跳ね返している。ひっじょーにくやしいのである。うーん，アツい。

ルールはわかりやすく，とてもとっつきがいい。それなりによくできたゲームであると思う。ゲーム自体が戦略性よりも技巧性に重きをおいているので，対戦よりはむしろ対コンピュータ戦がアツいのだけど。

とにかくしっかりできた佳作ゲームだ。

### ゲームはいいんだけどねぇ

かわいーもの系を狙っているのはよくわかる。実際，大きくなったときのプレイヤー1の白ウサギはぬいぐるみのようでしっかりそのセンになっている。ルールのとっつきもいいのでゲームとして，遊んでいて楽しいゲームだと思う。

が，ゲーム自体は面白くてかわいいのに，ほかの部分でかなり損しているのでは？

たとえば，あのパッケージイラスト。マニュアルの中のイラストのほうがよっぽど雰囲気にあっているではないか。

それに音楽。1曲1曲はいい曲だと思うが使い方を間違えてはいないか？ 特にセレクト面とゲーム中はどうにかすべきだ（雰囲気があっているのはスペシャルステージぐらいだ）。

狙いは非常によかった。だから，次はこういう細かいところも気をつけるべきだと思う。

| 総合評価 | 0　5　10 |
|---|---|
| ルール | ★★★★★★★★ |
| うるる～ | ★★★★★★★★ |
| 対コンピュータ | ★★★★★★★ |
| パッケージ | ★★★★ |
| お勧め度 | ★★★★★★ |

# 作戦名オーバーライド発動

「サイバーコア」はPCエンジンから移植されたシューティングゲームだったけど，この「ラストバタリオン」も同じくPCエンジンからの移植。もともとは「オーバーライド」という名前だった。知っているかな？

Yaegaki Nachi
八重垣 那智

私はパッドというものが苦手なので，家庭用ゲームというものが苦手である。なかにはRPGのように，機敏な操作を求められたりしないものもあるが，左手の親指が不器用なのは遺伝だということで，自分を納得させているのが現実である。

しかし，このラストバタリオンの原作である「オーバーライド」を初めて見せられたとき，私はいてもたってもいられなくなり，パッドであるにもかかわらず最後までプレイしてしまったのである。そういった意味では「オーバーライド」は私が人生の壁を乗り越えた，記念すべきゲームだったのである。

## 叩け人類の敵◆◆◆◆◆◆◆◆◆◆◆◆◆

操作はどこかで見たような8方向移動と攻撃＋スピードチェンジになっている。わかりやすいだろう。攻撃はボタンを押しっぱなしにするとフルオートになっているが，一定時間オフにすることでエネルギーを充填し，メガスマッシャーという波動砲攻撃も可能になっている。これは敵の弾やレーザーを消すことができるので，かなり多用することになる。また，スピードチェンジの瞬間に出る噴射で敵を攻撃することも可能だ。

そして自機のパワーアップであるが，Pのアイテムはメインショットのアップで，これは2段階までアップする。オプション兵器のほうはやや複雑になっていて，最後に取ったカラーボールの色で武器の種類が決まり，取っただけでその色が1段階アップする。つまりオプションの色それぞれが，別々にパワーアップしていくのだ。私は赤(斜め攻撃)と緑(正面攻撃)を集中的にパワーアップさせることをお勧めする。

自機はダメージ式で3枚のシールドを持っている。4発目が当たるとやられてしまうが，Eのアイテムでシールドを回復させることも可能だ。やられてしまった場合はすべてのパワーアップが失われるので，続けてダメージをくらわないように注意するべきだろう。もちろん自機を全部失うとゲームオーバーである。

赤と緑をパワーアップしよう

X68000用　5"2HD版2枚組　8,800円(税別)
スティング　☎03(3838)0433

## 移植アマイかショッパイか?◆◆◆◆◆

ゲーム展開は惑星ウラノスを舞台にし，宇宙空間から目標であるマザーコンピュータ「フレイア」まで，全6ステージで展開していく。原作の「オーバーライド」とは異なっていて，X68000での独自性を出そうとしているのを感じることができるのだが，どうも空回りしてしまったようにしか見えない。

たとえば，1面でとにかくデカキャラが出てくるのはいいが，それがどれもやたらと硬い。必然的に波動砲を使っていかなければならないのだが，充填中はこちらの攻撃ができなくなり，敵を攻撃し圧倒するシューティングゲームの醍醐味が減ってしまうことになる。

ボスキャラは結構デカイ

しかも，最初の面からこれでは，見た目にも難しく感じられてしまうのではないだろうか？　しかし，そういった点を乗り越え，先に進むことができれば，難易度もそう高くないのでシューティングとして遊べることは確かである。

ゲームセンターの縦スクロールシューティングゲームは，まずほとんどが縦画面である。それらがX68000に移植された場合，なんらかの制約が生じてしまうことは，いままでの前例から事実として受け止めることができる。

そこで，あらかじめ横画面を前提として作られた家庭用ゲームならば，感覚やシステムを損ねることなく移植できる。なおかつ，サウンドやグラフィックを強化できるという考えが生まれてくるのは自然なことだろう。

しかし，果たしてその移植に耐えうる作品があるか，そしてX68000ならではの特色を出すことができるかどうかは，別の問題として考えなければならないのではないだろうか？　そんなことを考えながらエンディングをぼんやりと眺めてしまう私であった。

### 硬派であれ

シューティングゲームは硬派でなくてはいけないと思うので，あまりごてごてとしたビジュアルシーンをつけてほしくないのが正直な感想だ。しかしマニュアルの人物紹介とかでは，いろいろと設定されているのだから，ゲーム内容にも生かしてほしいという気もする。だからといって内容が同じで，自機が空飛ぶ女の子になったら，それはそれでイヤだなぁ。うんうん。

| 総合評価 | 0　5　10 |
|---|---|
| ゲーム性 | ★★★★★★★ |
| 技術 | ★★★★★★ |
| 移植度 | ★★★★★★ |
| ビジュアル | ★★★★★ |
| デカキャラ固さ | ★★★★★★★★ |

# 相手のじゃまか,自分の積みか

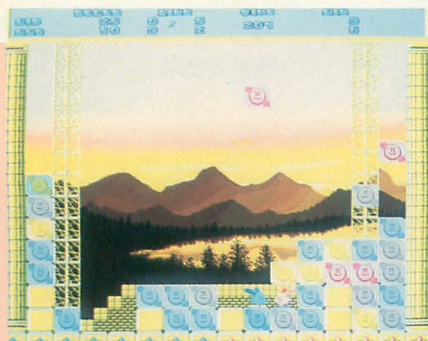
Mizuno Kazuo
水野 一雄

「テトリス」や「コラムス」のようなブロック型ゲームと,「ボンバーマン」のような対戦じゃまし合い型ゲームをドッキングさせたのが,この「PITAPAT」というゲーム。燃える要素の相乗効果はいかに？

私がOh!X編集部の幽霊ライターといわれている水野です。9月にとうとうX68000 XVIを手に入れました(初めてのX68000,それまではX1turboZだった)。メモリも8Mバイト実装しているので,湯水のようにメモリを使えて非常に快適です。プライベートのことはここまでにして,本題に入りましょう。

## いろいろなブロック◆◆◆◆◆◆◆◆◆

ルールは簡単。「PITくん」と「PATちゃん」と名付けられたキャラクター(ガンダムに出てきたハロを三角形にしたような形をしている)を操作して,「ジャンプ」「左右移動」「ブロックを消す」「ブロックを押す」の動作を行い,落ちてくるブロックの色の同じものを縦,横,斜めに4つ以上揃えて消していき(ブロックの消え方はコラムスと同じ),1ラウンドごとに指定されたライン数を消せば1ラウンドクリア。ブロックは6色あり,出現率の低いものほど高得点。その他,アイテムも出現する。

また,3ラウンドごとにラウンドセレクトがある。このときにはお城の絵が出てきて,右か左のどちらに進むかを決定できる。こうして,全64ラウンドのうち18ラウンドをクリアすれば1ゲーム終了となる。

左半分が青,右半分が赤の陣地

X68000用 5"2HD版2枚組 6,800円(税別)
ビクター音楽産業 ☎03(3423)7901

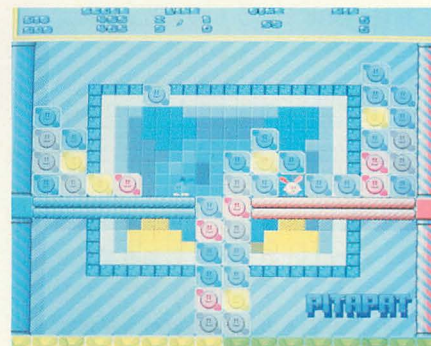
## とりあえずひとり◆◆◆◆◆◆◆◆◆◆◆

ディスクを入れてリセット。ファンタジックなBGMが流れ,タイトルへ。そばに誰もいないのでひとりでプレイしてみる。色とりどりのブロックが容赦なく落ちてくる。「PITくん」を操作して,ブロックを揃えようとするが,なかなか思うようにブロックが消せない。ジャンプしながらブロックを消すと,消してはいけないものまで消してしまう。ボタンを押すタイミングが難しい。ようやく,コツを掴んで調子に乗ってきた。「なんだ,こんなもんか」と思い始めるやいなや,「PATちゃん」が現れ,手当たりしだいにブロックを消し始めた。こうなると高得点を狙うのは難しい。とりあえずラウンドクリアを考えるが,ほどなく制限時間になってしまいゲームオーバー。何も考えず1ラインずつ消していては,1ラウンドクリアも難しい。再度チャレンジ。今度はじっくり考えてブロックを積み,一気に消去。じゃまものが現れる前にクリアした。

テトリスやコラムスではプレイヤー自身の能力(思考力や反射神経など)がゲームに反映されるが,このゲームではコンピュータ(PATちゃん)がブロックを消してしまうため,いくら考えてブロックを積んでも無駄になってしまうこともある。このため,ゲームオーバーになったときの悔しさがまったく違うのである。そして再プレイをする,しないはその悔しさによるのではないだろうか。

## やっぱり2人プレイ◆◆◆◆◆◆◆◆◆◆

対戦型なんだから2人で遊んでこそ,このゲームの面白さがわかるというもの。

「よし,よし,あと2ラインでクリアだ」
「そうは問屋が卸すかい」(相手の陣地に入って,3つ揃っているブロックを消す)
A「オー,マイガッ!」
B「どうだ,思い知ったか」
A「おしおきだべ〜」(アイテムを取って相手のブロックを消す)
B「ああっ,お許しくだせぇ,お代官様」
A「お主も悪よのう,越前屋」

とまあ,こんな調子で(どんな調子ぢゃ)楽しく遊べる(相手に対して腹が立つかもしれないが)。こちらは,人と人との駆け引きなので,2人で協力して高得点を目指すのもよし,お互いじゃまし合って共倒れもよし,と結構熱くなれる。

「対戦型アクションパズルゲーム」ということだけど,そのとおり,ぜひとも2人でプレイしてもらいたい。まあ,価格も手ごろだし,キャラクターも可愛いし,恋人となかよく遊ぶにはいいのではないかな。

ブロックに囲まれると消しまくらないとダメ

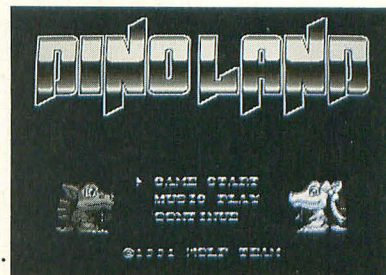
### カラフルな絵と音

X68000を手に入れて日も浅いので,BGMやグラフィックについてはよくわかりません(ごめんな〜:間寛平調)。私としてはいいと思うのですが(X1と比較してしまう)。BGMはOPMDRVを使用せず,オリジナルのドライバを使用しています。グラフィックは512×512モード。マップ中にCGが隠されているので,それを探すのもいいでしょう。

| 総合評価 | 0 — 5 — 10 |
|---|---|
| パズル度 | ★★★★★★ |
| 1人遊び度 | ★★ |
| 2人遊び度 | ★★★★★★★★★ |
| BGM | ★★★★★★★ |
| グラフィック | ★★★★★★★ |
| 価格 | ★★★★★★★ |
| 総合 | ★★★★★★ |

# 恐竜の卵，ならぬ恐竜の玉

Kageyama Hiroaki
影山 裕昭

台に向かって立ち，指を小刻みに動かす。目はボールを追っかけ，足はリズムを取る。ピンボールは定番中の定番のゲームだから，パソコンでも楽しみたい。この「ディノランド」は少し変わりダネだけどね。

ピンボールゲーム，なんだか懐かしい響きだなあ。いまではゲーセンでもあまり多く置かれていないようだけど，10年ほど前は近所の駅前にピンボール台だけのゲーセンがあって，電車の待ち時間によく遊んだものだ。ピンボールの面白さって，なんだろうとふと考えてみたけど，よくわからない。強いていえば，ボールを狙った位置に飛ばしたときの満足感かな。あとは，ピンボール台をところせましと走り回るボールのスピード感で興奮するのかもしれない。

パソコンでもこれまでいろいろと発売されたが，このゲームはめずらしくコミカルタッチのピンボールである。コミカルだから，女の裸は出てこない。うーむ，残念。おっと，そんなことはどうでもいい。なにがコミカルタッチなのか？ というと，ピンボール台の上に原始の世界を造り上げてしまって，台の上に置かれているターゲット（ボールを当てる的）も恐竜にしてしまったわけ。この恐竜も顔は怖いんだけど迫力がなくて，なんだかかわいい。

ゲームはピンボールシーンとボスシーンの2種類のステージで構成されている。ピンボールシーンは普通にフリッパーを操ってボールをパコパコと打っていればいい。このピンボールシーンで，ある条件を満たすとボスシーンに移る。ボスシーンでは，ボールを恐竜に変身させることができる。なんと，ボールまでも恐竜だったのだ。さらに，このボールならぬ恐竜には「ディノ」君という名前までついている。ボスステージではこのディノ君の恋人，「フレンディ」が登場する。フレンディは敵に囚らわれの身であったのだ。悪党に捕まった恋人を助け出す，このゲームは恐竜の愛の物語だったのだ（ウソだろ，おい）。

普通のステージ。コミカルな恐竜が並ぶ

## 遊んでみました◆◆◆◆◆◆◆◆◆◆◆◆

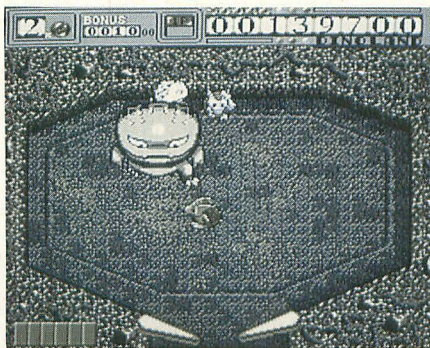
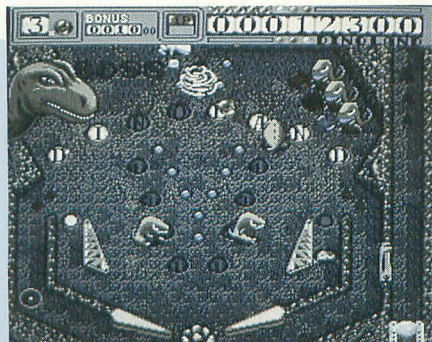
操作方法を詳しく説明しよう。キーボードで遊ぶ場合は，テンキーの6が右フリッパー，←キーで左フリッパーを操作する。OPT.2はボールの打ち出し，OPT.1はピンボールシーンでは台揺らし，ボスシーンではボール/恐竜の変身スイッチになる。

次にボスシーンの話だ。ボスシーンではボスキャラと，その手下（？）の敵キャラが出てくる。手下らしき敵キャラがフレンディに触れると，強制的にピンボールシーンに戻されてしまう。これを防ぐには恐竜に変身して，恋人に近づく敵キャラに体当たりすればいいのだ。こうすると敵の進行方向を逆にすることができる。恐竜になっていられる時間はわずかなので，素早く恐竜を操作しないとフレンディが奪われてしまうぞ。このすきにボールに変身してボスキャラに体当たりする。こうしてボスキャラを倒すと，1ステージクリアである。いい遅れたが，ピンボールシーンは1ステージは陸，2ステージは空，3ステージは海となっていて，ステージごとに台のデザインが変わることを付け加えておこう（海の次は陸に戻る）。

ボスシーン。うまく変身を使え

台の大きさは縦に2画面分で，ボールの動きに合わせて上下にスクロールする。ボールの動きは決して速いとはいえず，ピンボールゲームのスピード感を味わうにはかなりもの足りない。正直いって少々だらだらとしたゲーム進行であまり面白くない。また，ボールとフリッパーの衝突判定がいい加減で，台を揺らしたときなどボールがフリッパーを突き抜けるシーンも見られた。これにはア然としたが，うまく使えば落ちた玉を復活させることもできる。こういったバグは有効に使わせてもらおう。

あっそうそう，いい忘れていたことがあった。このゲームではMT-32系のMIDIにも対応している。ウルフチームの音楽が好きな人はチェックしておくといいかもね。

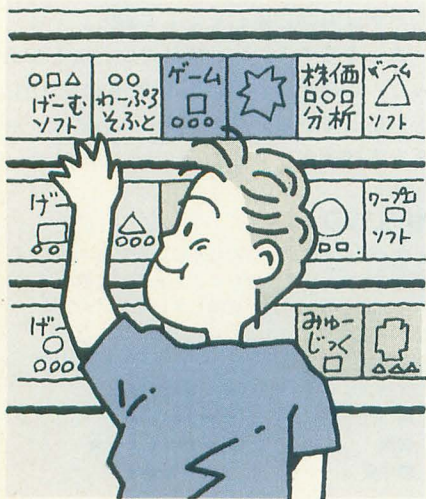
X68000用 5"2HD版2枚組 7,800円(税込)
ブラザー工業(TAKERU) ☎052(824)2493

### ピンボールは爽快なもの

とにかくボスシーンが難しい。下手をするとボスシーンで5秒ともたないこともある。そこで失敗すると，強制的にピンボールシーンに戻されてしまいちょっと悲しい。少なからず苦労してボスシーンに行っているんだから，せめてボスシーンの最初からやり直せるとかしないと，ゲームがなかなか先に進まずプレイヤーに飽きられてしまう。また，ピンボールゲームなのに，ハイスコアが残らないのも気に入らない。友達とワイワイと遊べないんじゃないかなあ？ ほかにもいいたいことはたくさんある。正直いって私にはこのゲームの目指しているモノが見えなかった。

**総合評価**（0〜10）

| 項目 | 評価 |
|---|---|
| グラフィック | ★★★★★★ |
| 操作性 | ★★★★★★ |
| 音楽 | ★★★★★★★★ |
| 熱中度 | ★★★★★★ |

# AFTER REVIEW

**オリジナルに忠実な移植，突っ拍子もないゲーム内容で話題を呼んだ「パロディウスだ！」。ノンストップで繰り広げられるパロディワールドに，ずっぽりハマッてしまう人が続出したゲームだ。**

## パロディウスだ!

▶理屈抜きに燃えます。そして登場する魅力的なキャラクターたちがとてもいい。「R-TYPE」とかの不気味な生物の敵はあまり見たくないけど「パロディウスだ！」の多様かつ強烈な敵は目が離せないんです。
海野　晶子(20)長野県

▶わっはっは，やっと「パロディウスだ！」のエンディングを見ることができました。いちばん難しかったのは「もっとも北の国から'90」です。やたら敵の出す弾には泣かされました。でも，いまは満足。
中島　慶洋(30)兵庫県

▶たくさん敵が出てくると，少し重くなるのが気になりますが上々の出来でしょう。グラフィック，音楽はいいし自機のバリエーションが多くて楽しい。
平田　昭夫(19)京都府

▶理由なんかない。いいものはいいのだ！
鹿又　健(22)栃木県

▶「パロディウスだ！」は感激もんである。「パロディウスだ！」はユーザー必携である。「パロディウスだ！」は新しい標準である。「パロディウスだ！」はまるで魔法である。「パロディウスだ！」は凄いぞ。
柳井　敏彦(32)愛媛県

▶ど～も，「パロディウスだ！」が難しいと思ったら1面でスピード3つ取り，2面からはパワーアップカプセルが余ればシールドつけてたもんなあ。どおりで5面から打ち返しがくるわけだ。
小林　直志(21)新潟県

▶2カ所ほど不満があるけど，そんなものはふっ飛んでしまうくらい「パロディウスだ！」は面白い。もうあっぱれというしかありませんねえ。今年のシューティングゲームはこれで決まり。こうなるとコナミさんの次回作が楽しみですね。フレーフレー，コナミ！
井上　博嗣(22)三重県

▶ゲームをやってX68000を買ったときのように胸がときめいたのはこのゲームだけ。
石田　正弘(21)北海道

▶「パロディウスだ！」は最高ですよ。文句のつけようがない出来だと思います。特にあの拡大縮小はどうやっているんでしょうかね。
安岡　毅(17)京都府

▶買ったときにはイロモノゲームだと思ったけど，やってみるとよく作ってあると感心しました。コナミ万歳！
阿妻　靖史(19)東京都

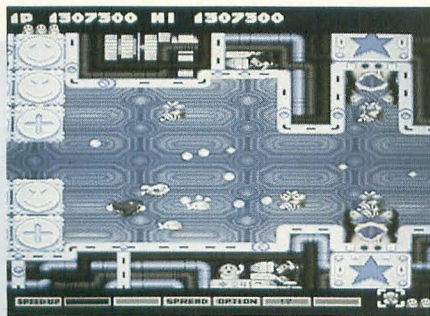
▶「パロディウスだ！」はいいですね。おかげで財布の中は寂しくなりましたが……。音楽よし，グラフィックよし，バランスもとてもいい。もういうことなしです。あたかもゲームセンターで遊んでいるような錯覚をしてしまうほどです。ちょっとほめすぎのような気がしますが，楽しいんだもん。
上池　宏幸(16)滋賀県

▶拡大縮小，ラスタースクロール，スーパーBGM，スーパーグラフィックの「パロディウスだ！」に誰も文句はいえないだろう。
永島　政信(16)群馬県

▶「パロディウスだ！」は凄い，凄すぎる。音楽もグラフィックも拡大縮小操作もなかなかよい。おまけにコナミのビデオに出てくるタイトルにはまいった。さすがコナミ。
原田　謙(16)広島県

▶それにしてもヨシオとヨシコがH。そして僕は7面のボスが好き。でもクリアしてから前に出ると死んでしまう。悲しいぞ。コナミの看板にも潰されたし。ちょっとまぬけかな。
羽田　直樹(19)東京都

▶やっぱりアーケード版との区別がほとんどつかないほど，完全移植されているからね。
木村　匡志(19)千葉県

▶いままでいろいろなシューティングゲームをやったが，笑えるシューティングゲームは初めてだ。曲もクラシックのアレンジでとてもいいしね。
植木　正幸(22)神奈川県

▶いままでのシューティングゲームはプレイする面白さだったけど，「パロディウスだ！」は見る面白さがあっていいと思う。

秋田　直人(17)宮崎県

▶私はやっぱりネコ戦艦が好きです。あ，チチビンタリカも捨てがたいな。

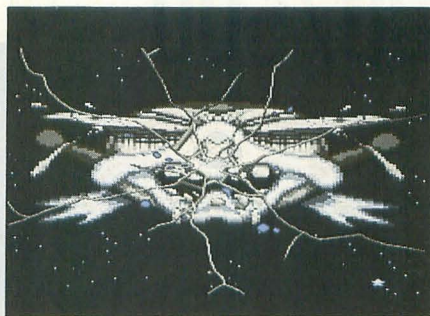
岡部　誠(26)福井県

▶小馬鹿にしたようなグラフィック，わけのわからんトラップなど，下手をすればただのまねっこパロディゲームになるだろうにコナミはよくこの内容をまとめたと思う。移植に関しても作り手のパワーが十分伝わってくるハイレベルな移植だし，いうことありませんね。いいぞ～。

田中　純一(18)鳥取県

▶え～ん，ネコ戦艦が倒せない。だってかわいすぎるんだもん。撃たれるたびに「ミューミュー」鳴くし，心を鬼にしても……やっぱりだめ。間違って倒してしまったときには3秒間の黙禱を捧げる私でした。

金子　美穂(16)埼玉県

▶私はシューティングゲームが苦手でいつも途中で投げ出してしまいます。しかし，「パロディウスだ！」はかわいいキャラターたちのオンパレードで，楽しく最後までプレイできました（コンティニューやりまくりで）。珍しく女性でも楽しめるゲームで私はとても満足しています。

坂井　紀子(18)宮城県

▶画面上に繰り広げられる数々のギャグの嵐。プレイ中はESCキーを多用して，ゲームを一時停止して思いっきり笑かしていただきました。う～ん，ゲーム性もいいし，音楽も最高。それにしてもブタ潮のフンドシ攻撃にはまいったなあ。看板にも潰されたし。コナミさん，次回もがんばってください。

山田　輝明(21)石川県

▶アーケードではプレイしたことがないので移植の出来について，あまりいえませんがX68000用のゲームとしては，最高の出来だと思います。今年のGAME OF THE YEARは「パロディウスだ！」で決まりですね。

佐藤　明敏(22)神奈川県

▶ただいま2周目にチャレンジ中。くっそ～，難しすぎる。ただでさえ厳しい攻撃に打ち返し弾なんか吐き出すんじゃねえ。絶対2周目をクリアしてやる。

鈴木　浩(15)東京都

◀桑田義久　千葉県

◀姉帯寛　神奈川県

## 発売中のソフト

★飛翔鮫　金子製作所
X68000用　5″2HD版　8,800円(税別)

★プロサッカー68　イマジニア
X68000用　5″2HD版　9,800円(税別)

★出たな!! ツインビー　コナミ
X68000用　5″2HD版　9,800円(税別)

★ゼノン2　エピック・ソニー
X68000用　5″2HD版　9,800円（税別）

★アルシャーク　ライトスタッフ
X68000用　5″2HD版　9,800円(税別)

★コード・0　エニックス
X68000用　5″2HD版　7,800円(税別)

## 新作情報

★ヴェルスナーグ戦乱　ファミリーソフト
X68000用　5″2HD版　9,800円(税別)

★スターウォーズ　ビクター音楽産業
X68000用　5″2HD版　7,200円(税別)

★ノア　M.N.Mソフトウェア
X68000用　5″2HD版　7,200円(税別)

★大戦略III'90　システムソフト
X68000用　5″2HD版2枚組　9,800円(税別)

★スーパー上海ドラゴンズアイ　ブラザー工業(TAKERU)
X68000用　5″2HD版　7,800円(税込)

★SPINDIZZY II　アルシスソフトウェア
X68000用　5″2HD版　8,700円(税別)

★PITAPAT　ビクター音楽産業
X68000用　5″2HD版2枚組　6,800円(税別)

★ジェノサイド2　ズーム
X68000用　5″2HD版　8,800円(税別)

★シムアース　イマジニア
X68000用　5″2HD版　12,800円(税別)

★レミングス　イマジニア
X68000用　5″2HD版　9,800円(税別)

★F29 RETALIATOR　イマジニア
X68000用　5″2HD版　価格未定

★ワールドゴルフIII　エニックス
X68000用　5″2HD版　9,800円(税別)

★ヘビーノヴァ　マイクロネット
X68000用　5″2HD版　7,800円(税別)

★ウェルトリス　BPS
X68000用　5″2HD版　7,800円(税別)

★ファーストクイーンII　呉ソフトウェア工房
X68000用　5″2HD版　8,800円(税別)

★エイリアンシンドローム　電波新聞社
X68000用　5″2HD版　価格未定

★ブルトン・レイ　シナリオ集VOL.3　システムソフト
X68000用　5″2HD版2枚組　価格未定

★ユニオン　ポニーテールソフト
X68000用　5″2HD版　価格未定

★ふしぎの海のナディア　ゼネラルプロダクツ
X68000用　5″2HD版　価格未定

★伊忍道・打倒信長　光栄
X68000用　5″2HD版　9,800円(税別)

★究極タイガー　金子製作所
X68000用　5″2HD版　価格未定

**製品紹介**

GS規格対応音源モジュール

# CM-300/500

Kioi Makoto 紀尾井 誠

**Rolandから発売される新しい音源モジュールCM-300とCM-500の概要を紹介しましょう。どちらもGSフォーマットに沿った製品です。どうやらGS規格もSC-55だけで終わるようなことはないようですから，とりあえずひと安心というところでしょうか。どちらも低価格に設定されているので今後の普及が期待される製品です。**

## CM-300

CM-300をひと言でいうと，「SC-55から操作パネルやリモコンをなくして値段を下げたもの」となります。液晶表示など機能的には多少削られたものの，音源としての性能ではまったく変わりないといっていいでしょう。MT-32互換音色とドラムセットもバリエーション127番として残されているようです。ただし，SC-55のように互換モードで起動することはできませんのでオマケの装備と考えておいたほうが無難です。

試用してみても，音源はSC-55と同じものが使われているようで，まったく同じ音が出ます。これはGS規格で規定されているメインキャピタルの音色に限らず，バリエーション部も同様でドラムセットの構成もまったく同じです。

エクスクルーシブ関係はどうでしょうか。CM-300およびCM-500のGS音源部分はモデルIDとして$42のみが与えられています。これは広くGS音源全体がサポートすべきIDです（SC-55はGS音源としての$42のほかSC-55専用の$47を持っている）。

懸念されたエクスクルーシブによる音色変更などもまったく同じ状態で使用できるようですし，コントロールチェンジでのNRPNによる音色エディットもそのまま踏襲されています。GSフォーマットで作成するデータではすべてのコントロールチェンジと$42で定義されたすべてのエクスクルーシブメッセージを使用しても互換性は保証されるということでしょう。楽器ごとに拡張された部分があれば，それは別のモデルIDで定義されている，というわかりやすい構成に落ち着いたようです。

SC-55とのデータレベルでの互換性は完璧と考えてもいいでしょう。

内部ハードウェア的にはほとんど同じなので，外側に話を移します。SC-55のAUDIO IN端子は踏襲されました。X68000本体音源やほかの音源とミキシング可能です。オーディオまわりの接続はすべて標準オーディオプラグで行われます（SC-55はRCAピンジャック）。

CM-32Lと同じような形をしていますが，MIDI RECIEVEのLEDがAUDIOに変更されています。

## CM-500

CM-500をひと言でいうと，「MT-32とSC-55の音源部分をひとつの箱に入れたもの」となります。これまでのDTMの中心だったLA音源と今後の発展が期待されるGS音源の両方が使えるわけです。これらは単に音源だけでなく，エクスクルーシブのレベルでも高い互換性を持ったものです。

LA音源部はCM-32Lに同じ，GS音源部はCM-300と同じと考えてかまいません。

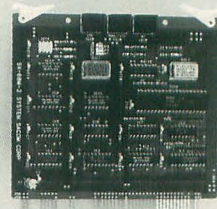
さらに，「CM-64エミュレートモード」を備えています。これはCM-64を構成するCM-32Pの部分をGS音源で代用させるモードです。現時点ではCM-500の実機が手元にないため十分に確認しているわけではありませんが，噂ではかなり使いモノになるということです……。もちろん，RS PCMカードを使ったもの（カードスロットがない）やエクスクルーシブを多用したもの（一部はサポートされている）は駄目でしょうが，極端なクセのないPCM系の音色ならSC-55の音源でも無理がないと考えられます。SC-55の「MT-32エミュレートモード」よりは安心して使えそうです。

音色はCM-32Pの内蔵64音のほかに，よく使用するカードの音色からセレクトされたものが64音追加設定されています。念を

CM-500

### MIDIボードSX-68M-2

システムサコム製のMIDIボードが新しくなった。従来品は標準MIDIケーブルを直接接続するようになっていたため，スペースの関係でSCSIボードなどの一部のボードと併用するのが難しいことがあった。今回の製品はシャープ純正品と同じく，一般のMIDIケーブルとは中間ケーブルを用いてボードに接続する方式を採用した。これにより，従来の本体へのはみ出しがなくなった。

MIDIコントローラは純正品とまったく同じで，ソフト的な互換性はまったく問題がない。ただし，従来品同様テープシンク端子は付属していない（使う人はまずいない）。価格は19,800円と純正品に比べて割安になっている。

押しておきますが，セレクトされたのはあくまでも音色名ですから，鳴るのはGS音源部分の音色です。

まあ，どうしてもRS PCMカードが必要というときはCM-32Pを増設すればよいわけです。これは20万円以下でできる最強の取り合わせといえるでしょう。

さらにCM-500ではMT-32と同等品を内蔵しながら，GS音源部にはMT-32互換音色を内蔵するという万全の体制をとっていますから，CM-500+CM-32Pという構成をとればMT-32/CM-64/SC-55で考えられるあらゆるデータが演奏できることになります。

## CM-500の4つのモード

さて，CM-500には4つのモードがあります。

**●モードA**

PC-9801でミュージ郎Ver.2.0を使うためのモードです。

GS音源は1～16チャンネルのすべて，LA音源は2～10の9チャンネルを使用します。要するに2台の音源をMIDI THRUでつないだのと同じでなんの細工もせずにすべての音源が一斉に鳴るモードです。マスターボリュームは，

GS=127

LA=100

に初期設定されます(要するにLA音源部分は少し音量を絞ってある)。2～10チャンネルは信号を送ると両方の音源が鳴り出しますから，使い分ける場合はエクスクルーシブメッセージで双方の音源のチャンネル設定をON/OFFしなければなりません。しかし，すべての音源を使い切ることのできるモードでもあります。

**●モードB**

CM-64エミュレートモードです。

GS音源はチャンネル11～16，LA音源はチャンネル2～10に割り当てられ，GS部のパンポットは反転します。マスターボリュームは双方とも100に設定されます。

LA音源についてはたいていのエクスクルーシブをはじめ，どんなデータでも忠実に演奏します。PCM音源の代わりをGS音源がやっているのですが，CM-32Pの最大発音数は32パーシャル，SC-55では24パーシャルというハードウェア上の制約があります。その範囲内でCM-32Pのエクスクルーシブメッセージの一部を認識し実行します。たとえば，エクスクルーシブによるパーシャルリザーブなどは32パーシャル用のデータを24パーシャル用に自動変換するようです。

**●モードC**

GS音源としてのモードです。

チャンネル1～16のすべてがGS音源となります。LA音源は動作しません。

**●モードD**

キーボードなどに接続して使用するときのモードです。

GS音源にチャンネル1～10，LA音源にチャンネル11～16が割り当てられています。マスターボリュームは双方127です。LA部分のパンポットも反転されています。2台の楽器を使うときの見本のようなセッティングです。ただし，チャンネルによって使用できるコントロールチェンジが違うので，2台分の楽器だということには留意しておかなければなりません。

モードAのようにエクスクルーシブを多用することもなく，双方の音源を余力を持って使用できるモードです。チャンネル10(ドラムパート)にはGSのものが使われているのもポイントでしょうか。

ミュージ郎とは縁のないX68000では，CM-500専用のデータを作成する場合にはもっとも使いやすいモードかもしれません。

＊　＊　＊

今後はCM-500対応の曲データというのも現れてくることでしょう。そういった場合，すでにMT-32(CM-32L/CM-64)やSC-55を持っている人なら他方の音源を増設することで，CM-500とほぼ同じ環境を設定できます。その際に前記のような設定を行えば(手作業で)，各モードに対応したデータが再現できるはずです。

ハードウェアの外回りはCM-300とほぼ同じです。相違点はAUDIO INは付属してないこと，モード切り替えスイッチが背面についていることくらいです。

## どれを選ぶか?

ここで，Rolandのモジュールをまとめてみましょう。別にRoland社の音源でなければMIDIはできないというわけではありませんが，手頃な値段で，パソコンからの応答も高速なものというと，どうもこのシリーズしか挙がってこないようです。これらについて多少独断と偏見も交えて解説しておきましょう。

まず，

| | |
|---|---|
| CM-300 (GS) | 58,000円 |
| CM-32L (LA) | 69,000円 |
| SC-55 (GS) | 69,000円 |
| CM-32P (PCM) | 72,000円 |
| CM-500 (LA，GS) | 115,000円 |
| CM-64 (LA，PCM) | 129,000円 |

値段順では以上のようになります。

### パンポット

MT-32，CM-32L/64などではパンポット(音像の左右定位。要するにステレオ出力の指定)が世間と左右反対になっていました。SC-55のMT-32モードではオーディオ出力の左右を入れ替えて対応するようになっていたわけですが，CM-500では各モードで自動的に対応します。0が左右どちら側になるかを表1に示します。

表1

| | LA | GS |
|---|---|---|
| モードA | 右 | 左 |
| モードB | 右 | 右 |
| モードC | 出力なし | 左 |
| モードD | 左 | 左 |

市販アプリケーションがLA音源に対応するのはほぼ間違いありませんから,LA音源のものを購入するのは無難といえます。特にCM-32LはMIDI対応のゲームをやりたい人や「とりあえずMIDIを始める」人には適しています。

CM-32PはRSPCM方式のカードが使えるPCM音源です。音質ではかなりよい評価を得ているようです。が，これを最初の楽器として購入する人はまずいません。これはCM-32Lと組み合わせることで，CM-64と同じ音源構成になりますので，CM-32Lの増設用といった雰囲気の音源です。とりあえずCM-32Lを買っておいて必要ならCM-64相当に増設することができるわけです。といっても,CM-32Pを増設したという話はほとんど聞いたことがありませんが……。

CM-64はMIDIを始めるにあたってCM-32Lでは少し不安があるという人には人気があるようです。現状ではほとんどのデータはCM-32Lだけでこと足りるのですが,「どうせなら……」ということでCM-64を買っていく人が多いと聞きます。最大で60音以上を演奏できる音源としては破格の値段といえるでしょう。最近はCM-64対応のデータも徐々に増えてきています。現在ではとりあえずCM-64とカードを何枚か持っておけば巷にあるほとんどのデータを演奏可能です。

SC-55はLA音源の音では満足できない人のための音源です。現在，コンピュータミュージックでLA音源が標準となっているのは動かしがたい事実です。しかし，はっきりいってしまえば，多くの人がもう少しマシな標準を求めていたのも事実です。そこへGSフォーマットというのを引っ提げてSC-55が登場しました。GSフォーマットはDTMを強く推進しているRolandが提唱している標準音源仕様です。

これからの標準になるかどうか市場の動向が注目されていますが，危険を冒したくない人はGS音源はまだ少し様子を見たほうがよいと思います。

また,GS音源はシンセサイザではありませんから，自分で音色を作るといったことはできません。あくまでも用意された音を使用することになります。ミュージックデータの互換性のためにはこのほうがいいのですが，好みにより意見が分かれるところでしょう。

CM-300は低価格なSC-55と考えてかまいません。MT-32/CM-32Lが全盛となった理由を考えると，なにより低価格だったことが考えられます。音のいい音源だったらほかにもあるのですが，値段的には倍以上のものがほとんどですから。CM-32Lに多くの人が音質的に割高感を持っているのに対して，SC-55ではほとんどの人が割安感を持っているというのも事実です。そういった意味ではSC-55やCM-32Lよりもさらに安価なCM-300は将来の標準として最右翼ともいえます。

CMシリーズでは操作部がまったくないというのは評価が分かれるところですが，自分でデータを作らないという人ならあまり関係ないでしょう。液晶表示部分は利用価値が高いものなのですが，約1万円以上安いというのはやはり大きいかもしれません。

＊　＊　＊

今回はCM-500本体がまだ入手できていないため，マニュアルから推察される概要しかお伝えできませんでした。特にCM-64エミュレートモードについては機会を見てまた紹介したいと思います。

図1

| PC# | 音色名 | V | PC# | 音色名 | V | PC# | 音色名 | V | PC# | 音色名 | V |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | A. Piano 1 | 2 | 33 | Choir 4 | 2 | 65 | Harpsichord | 1 | 97 | Timpani | 1 |
| 2 | A. Piano 2 | 2 | 34 | Strings 1 | 1 | 66 | Coupled Hps. | 2 | 98 | Jazz Gt. | 1 |
| 3 | A. Piano 3 | 2 | 35 | Strings 2 | 1 | 67 | Church Org. 1 | 1 | 99 | Hawaiian Gt. | 1 |
| 4 | A. Piano 4 | 2 | 36 | Strings 3 | 2 | 68 | Church Org. 2 | 2 | 100 | Mutet Gt. | 1 |
| 5 | A. Piano 5 | 1 | 37 | Strings 4 | 2 | 69 | Tinkle Bell | 1 | 101 | Chorus Gt. | 2 |
| 6 | A. Piano 7 | 1 | 38 | E. Organ 2 | 2 | 70 | Steel Drums | 1 | 102 | Overdrive Gt | 1 |
| 7 | A. Piano 9 | 1 | 39 | E. Organ 4 | 2 | 71 | Celesta | 1 | 103 | Distortion Gt | 1 |
| 8 | E. Piano 1 | 2 | 40 | E. Organ 6 | 2 | 72 | Sitar | 1 | 104 | Feedback Gt. | 2 |
| 9 | E. Piano 3 | 2 | 41 | E. Organ 8 | 2 | 73 | Santur | 1 | 105 | Gt Harmonics | 1 |
| 10 | E. Piano 5 | 2 | 42 | E. Organ 9 | 2 | 74 | Koto | 1 | 106 | Fantasia | 2 |
| 11 | A. Guitar 1 | 1 | 43 | E. Organ 10 | 2 | 75 | Pan Flute | 1 | 107 | Space Voice | 1 |
| 12 | A. Guitar 3 | 2 | 44 | E. Organ 11 | 2 | 76 | Piano 3 | 1 | 108 | Solo Vox | 2 |
| 13 | A. Guitar 4 | 2 | 45 | E. Organ 12 | 2 | 77 | Clav. | 1 | 109 | Metal Pad | 2 |
| 14 | E. Guitar 1 | 2 | 46 | E. Organ 13 | 2 | 78 | Violin 1 | 1 | 110 | Synth Brass1 | 2 |
| 15 | E. Guitar 2 | 1 | 47 | Soft TP 1 | 1 | 79 | Violin 2 | 1 | 111 | Synth. Strings 1 | 1 |
| 16 | Slap 3 | 1 | 48 | Soft TP 3 | 1 | 80 | Cello 1 | 1 | 112 | Synth. Strings 2 | 2 |
| 17 | Slap 4 | 2 | 49 | TP／TRB 1 | 1 | 81 | Cello 2 | 1 | 113 | E. Piano 2 | 1 |
| 18 | Slap 5 | 1 | 50 | TP／TRB 2 | 1 | 82 | Contrabass | 1 | 114 | Detuned EP 2 | 2 |
| 19 | Slap 6 | 2 | 51 | TP／TRB 3 | 1 | 83 | Pizzicato | 1 | 115 | Syn Vox | 1 |
| 20 | Slap 9 | 1 | 52 | TP／TRB 4 | 1 | 84 | Harp | 1 | 116 | Synth Bass 1 | 1 |
| 21 | Slap 10 | 2 | 53 | TP／TRB 5 | 2 | 85 | Oboe 1 | 1 | 117 | Synth Bass 2 | 2 |
| 22 | Slap 11 | 1 | 54 | TP／TRB 6 | 2 | 86 | Oboe 2 | 1 | 118 | Synth Bass 3 | 1 |
| 23 | Slap 12 | 1 | 55 | Sax 1 | 1 | 87 | Bassoon 1 | 1 | 119 | Synth Bass 4 | 2 |
| 24 | Fingered 1 | 1 | 56 | Sax 2 | 1 | 88 | Bassoon 2 | 1 | 120 | Synth Bass 2 | 2 |
| 25 | Fingered 2 | 2 | 57 | Sax 3 | 1 | 89 | Clarinet 1 | 1 | 121 | Organ 3 | 2 |
| 26 | Picked 1 | 1 | 58 | Sax 5 | 2 | 90 | Clarinet 2 | 1 | 122 | Alto sax | 1 |
| 27 | Picked 2 | 2 | 59 | Brass 1 | 1 | 91 | Clarinet 3 | 1 | 123 | Tenor sax | 1 |
| 28 | Fretless 1 | 1 | 60 | Brass 2 | 1 | 92 | Fr. Horn 1 | 2 | 124 | Baritone sax | 1 |
| 29 | AC. Bass 1 | 2 | 61 | Brass 3 | 2 | 93 | Fr. Horn 2 | 2 | 125 | Trombone | 1 |
| 30 | Choir 1 | 1 | 62 | Brass 4 | 2 | 94 | Fr. Horn 3 | 2 | 126 | Melo Tom | 1 |
| 31 | Choir 2 | 1 | 63 | Brass 5 | 2 | 95 | Tuba 1 | 1 | 127 | Synth Drum | 1 |
| 32 | Choir 3 | 2 | 64 | Orche Hit | 1 | 96 | Tuba 2 | 1 | 128 | 808 Tom | 1 |

PC#：プログラム・ナンバー
V：使用ボイス数

MT-32リズムパート活用講座

# LA音源を活用しよう

Tama Tamaki たま たまき

**「MT-32のドラムを強化したい」と思っている人は多いと思います。それにはまず，LA音源の構成から知らなければなりません。あまり行われていないリズムパートのエディットとサスティーンつきリズム音の設定についても解説します。**

最近，「MT-32のドラムがへこいっ！」という人が増えてきました。でも，そういう人に「MT-32で音を作ったことがあるの？」と尋ねると，「作れない」なんて答えが返ってきます。

確かにRS-PCMの「ロックドラムス」の音はMT-32のプリセットよりよい音だけど，ロックドラムのドラムキットはCMシリーズ上だとキーアサインを変更できないし，音を作るという点ではなにかと不便を感じます。

MT-32のリズムパートにアサインする音はシンセパートとまったく同じなので，本気で音色を作ればかなりの音を鳴らすことができるのです。ただし，若干の注意事項はありますが……。

ということで，ここではMT-32のリズムパートの活用法を解説します。ちなみにD110はもちろん，D10/20でもあまりやることは変わらないので参考になるでしょう。

## LA方式についておさらい

「LA方式を知らずして，MT-32で音は作れない」ということで，まずはLA方式のおさらいをすることにします。

**●パーシャル**

パーシャルとはLA方式では音を構成する最小単位です。これで基本的な音の性格が決まります。MT-32のパーシャルはだいたい図1のような構造になっています。

MT-32のパーシャルはシンセサイザサウンド系列とPCM系列に分かれていて，PCM系列を使用するときはTVF（フィルタ）を使用できない構造になっています。

そもそも，LA方式でのPCMは，アタック時の複雑な倍音成分を補うためのものであると考えられていたらしく，フィルタは必要ないと判断されたのでしょうか？ とにかくPCMではTVFは使えません。もっともこの点はD70で改善されましたけどね。

さて，一応各ブロックについて簡単に説明しておきましょう。

### 1.シンセサイザサウンド部

かなりの倍音を含んだ波形を発信する回路です。MT-32では矩形波(SQU)とのこぎり波を選択できます。

### 2.PCMサウンド部

内蔵のWAVE-ROMに記録されている波形を出力します。MT-32では128種類登録されていますが，CM-64/32Lではバンクが増設されていて，サウンドエフェクトを中心に128種類追加されています。

*　　*　　*

上記の2つのブロックの前にピッチというものがありますが，基本となるC4キーの音を指定します。これを基準に，キーボードから入力された（MT-32ではMIDIから入力された）音程の周波数が決まります。その音程にピッチエンベロープとLFOをかけることができます。

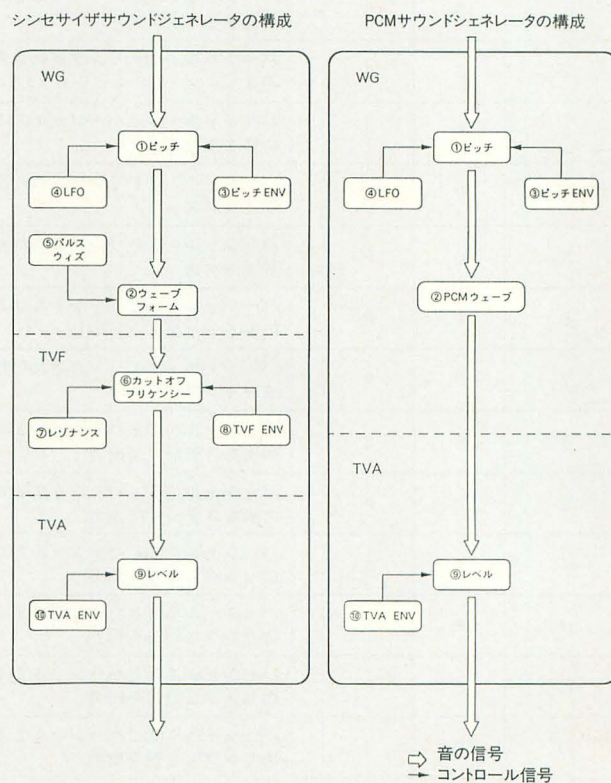
これは，上記2つのブロックと一緒に考えたほうがわかりやすいので，上記2つのブロックに含まれていると思ってください。

また，この2つをひっくるめてWG（ウェーブジェネレータ）と呼ぶことがあるので注意しましょう。

### 3.TVF(タイムバリアントフィルタ)

WGで出力された波形を，必要な部分のみ抽出（フィルタリング）する回路です。タイムバリアントとは時間経過により変化させることができるというような意味だと思います。エンベロープジェネレータがついているフィルタということですね。MT-32のTVFはローパスフィルタしかありません。ローパスフィルタとは，ある周波数（カットオフフリケンシー）で指定した周波数より低い成分を通過させるフィルタです。要するに高音部分を削るものです。

TVFにはレゾナンスという機能があります。これを使うとカットオフフリケンシーで指定した周波数成分の付近を強調し，

図1

音色に変化をつけることができます。TVFのエンベロープジェネレータはカットオフフリケンシーに対して作用します。

### 4.TVA(タイムバリアントアンプリファイア)

音量を調節する回路のことです。タイムバリアントとついているので当然，時間的変化（エンベロープ）をつけることができます。

## ストラクチャー

パーシャルで作られた音をどのような組み合わせで出力するかを指定するのがストラクチャーの役目です。

MT-32ではストラクチャーは13タイプ用意されており，パーシャル1と2，3と4が対になっています。

なかにはリングモジュレータという回路を通すものがあります。リングモジュレータとは一種の変調器で，簡単にいっちゃうと2つのパーシャルをかけ合わして多くの倍音成分を含んだ音にする回路ということです。

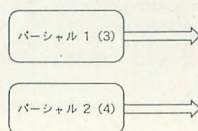
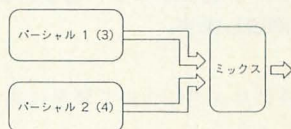
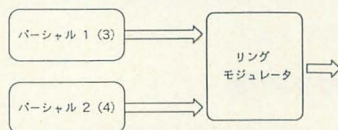
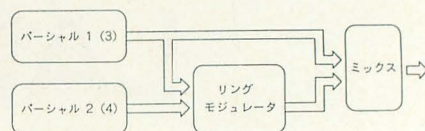
ストラクチャーは大きく分けて4つのタイプに分類できます。

パーシャル1はL側優先，パーシャル2はR側優先で出力されます。要するにパーシャル同士をミックスしないタイプです。

パーシャル1とパーシャル2をミックスした音が出力されます。これがもっとも基本形でいちばん使用頻度が高いタイプです。

パーシャル1とパーシャル2をリングモジュレータで変調をかけて出力します。

パーシャル1の音と，2つのパーシャルをリングモジュレータで変調をかけた音をミックスして出力します。ディストーションギターなんかはこのタイプで作るとよいのではないかと思います。

**図2**

S（シンセサイザサウンドジェネレータ）
P（PCMサウンドジェネレータ）

| ストラクチャーナンバー | パーシャル1 | パーシャル2 | パーシャルの組み合わせ方 | ブロック図 |
|---|---|---|---|---|
| 1 | S | S | パーシャル1(3)とパーシャル2(4)の音をミックス | S S |
| 2 | S | S | パーシャル1(3)とリングされた音をミックス | S S R |
| 3 | P | S | パーシャル1(3)とパーシャル2(4)の音をミックス | P S |
| 4 | P | S | パーシャル1(3)とリングされた音をミックス | P S R |
| 5 | S | P | パーシャル1(3)とリングされた音をミックス | S P R |
| 6 | P | P | パーシャル1(3)とパーシャル2(4)の音をミックス | P P |
| 7 | P | P | パーシャル1(3)とリングされた音をミックス | P P R |
| 8 | S | S | パーシャル1(3)とパーシャル2(4)の音をステレオで出力 | S S |
| 9 | P | P | パーシャル1(3)とパーシャル2(4)の音をステレオで出力 | P P |
| 10 | S | S | パーシャル1(3)とパーシャル2(4)のリングした音を出力 | S S R |
| 11 | P | S | パーシャル1(3)とパーシャル2(4)のリングした音を出力 | P S R |
| 12 | S | P | パーシャル1(3)とパーシャル2(4)のリングした音を出力 | S P R |
| 13 | P | S | パーシャル1(3)とパーシャル2(4)のリングした音を出力 | P P R |

## リズムパートの構造

さて，本題のリズムパートについての説明に入ります。少なくとも，リズムパートをセットアップするには，エクスクルーシブメッセージのリズムセットアップとトーンメモリを変更することになります。MT-32のトーンエディタがあれば，あまり考えなくてもよいのですが，X68000では，単独で市販されているトーンエディタはありません。Mu-1 Superのトーンエディタかフリーウェアのエディタを使うしかありません。エクスクルーシブをダンプで書いて直接送り込んでる人もけっこういると思います。

さてと，まずは比較的簡単なリズムセットアップのエクスクルーシブから説明しましょう。MT-32のリズムセットアップエリアはアドレス03 01 10から始まり，1キーにつき4バイトのデータを持っています。

OFFSET Description
+0 TONE 00H-3FH=internal,
40H-7EH=rhythm preset,
7FH =off
+1 OUTPUT LEVEL 00H-64H
+2 PANPOT 00H(L)-07H(C)-0EH(R)
+3 REVERB SWITCH 0=OFF,1=ON
D-110ではOUTPUT ASSIGN

ここで注意しなければならないパラメータといえば，PANPOTでしょう。コツとしては有名なアーティストのドラムセットやパーカッションの配置を思い浮かべながら設定するといいんじゃないかと思います。ビデオや写真を見て研究しましょう（音楽雑誌のノリだな）。

TONEにはPRESET A,Bは指定できませんが，INTERNALにコピーすれば使えるということも覚えておいてください。

意外な使い方としては，Elec BASSをC2からB3，Slap BASSをC4からB4にアサインして，リアルなチョッパーベースのフレーズをリズムパートで演奏させるとか，オケヒットをリズムにアサインして演奏させ

るなど，慣れてくればいろんなことができます。ただし，1つひとつキーアサインを書いていかなきゃならないのが難点ですが……。

あとのパラメータは解説するほどのものではないでしょう。

## リズムエディットのテクニック

リズムアサイン用の音色を作るに当たって以下のパラメータが注意すべきポイントです。よく覚えておくようにしてください。

### ■コモンパラメータ

**●ENVモード**

各パーシャルのENVでNOTE OFFを無視するかどうかを設定します。NO SUSに設定した音はNOTE OFFを無視します。といっても音が止まらなくなるわけではなく，厳密にいうとENVのサスティーンを無視します。つまりNO SUSでは音がブツ切りみたいになってしまうわけです。

D110のマニュアルによると「通常はNOMALにしておきますが，リズム音の場合はNO SUSに設定する」などと書いてありますが，NORMALのままでリズム音を作るのもまた一興でしょう（おまけデータのスネアの音に注目！）。

ただし，NOMALに設定された音をリズムパートで演奏させると（それだけ長く音源を占有するわけですから），ときには音切れの原因になることもあります。十分試してから使いましょう。

### ■WGグループ

**●ピッチコース，ピッチファイン，キーフォロー**

ピッチコースで音程，ピッチファインで微妙な音程差を指定します。

キーフォローとはキーコードの指定に従うかどうかの指定です。リズム音はキーフォローを0に設定します。なぜかというと0にしないと発音周波数が固定にならないからです。

キーフォローを0に設定しておけば，どのキーにアサインしてもピッチコースとピッチファインで設定した音が発音されるわけです。

また，ベンダースイッチをONにしておけば，リズムパートでもベンダー情報が効くようになるので，ピッチベンダーでポルタメント効果を出すこともできるのです。

## 参考のために

最後に私が愛用しているMT-32用のリズムキットのエクスクルーシブデータを付記しておきます。あまり詳しく説明できなかったので，あとは個人個人で解析するためのサンプルというわけです。マニュアルのMIDIインプリメーションのエクスクルーシブメッセージのところと突き合わせてみるとよいでしょう。

また，MK_EXC.C はテキストファイルで書かれたダンプデータをMIDIのチェックサムを計算しながらバイナリ形式に変換する簡単なユーティリティです。チェックサム計算が面倒くさいので手抜きで作ってみました。

まず，MK_EXC.Cをエディタで入力し，XCなどでコンパイルして MK_EXC.X を作成してください。

次にエクスクルーシブデータ MT_SY22D.TXTを入力し，コマンドラインでMK EXC.Xを実行します。

MK_EXC　MT_SY22D.TXT　MT SY22D.MDD

これでMT_SY 22 D.MDDというZMUSIC用のエクスクルーシブデータを作成します（普通のエクスクルーシブデータを16進ASCII文字にしたもの)。あとは楽器にデータを転送するだけです。ZMUSIC.Xを組み込んで，

A>COPY MT_SY22D.MDD MIDI

のようにします。その他のドライバをお使いの方は，

fputs(b_hexS(buf,～),fp2)；

の部分を，

fputc(～,fp2)；

のように変更し，0x0d，0x0aの出力部分を削除してください。これでMIDIの生データが生成されます。ちなみに，MUSICDRV.Xを使用している場合なら，

A>COPY MT_SY22D.MDD MIDIB

となります。

＊　　＊　　＊

最後に「よい音」というものは偶発的に生まれるものです。パラメータをちょろちょろいじらないことにはなにもできませんし，理解することもできないでしょう。実践あるのみ，これを機会に皆さんも音作りにチャレンジしてはいかがでしょうか？

図３　SUSTAINの違い

リスト1

```
 1: /*
 2:    MIDI exclusive data file make utility. Version 1.03.1
 3:    Programmed Mar. 25, 1991 by Tama-chan.
 4:    Arrenged for Z-MUSIC by Oh!X
 5:
 6:    書式一覧
 7:
 8:    & = sum on/off
 9:            バルクデータ内でブロックごとにチェックサムが必要なものが
10:            あるので、その部分をこいつではさみ込む。
11:    * = comment
12:            こいつが現われると行末(0ah)までコメントとみなし、無視する。
13:    $ =
14:            数値を16進数で記述するときに指定する。一度書くと、以下16
15:            進数とみなされる。デフォルト。
16:    # =
17:            数値を10進数で記述するときに指定する。一度書くと、以下10
18:            進数とみなされる。
19:    % =
20:            数値を2進数で記述するときに指定する。一度書くと、以下2進数
21:            とみなされる。
22:    0-9,A-F,a-f = data.
23:            言うまでもなく数値データである。and 255されるからね。
24:    etc...
25:            区切データとみなす。
26:    はっきりいって、なんて単純なプログラムなのだろう。ゲヘへ。
27: */
28:
29:    #include        <stdio.h>
30:
31: FILE        *fp1,*fp2;
32:
33: main(int argc, char *argv[])
34:
35: {
36:
37: int         i,j=16,c,s=0,sum=0,sw=0,co=0;
38: char        buf[10];
39:    puts( "MIDI exclusive data file make utility. Version 1.03¥n"
40:            "Programmed Mar. 25, 1991 by Tama-chan.¥n");
41:
42:    if (((fp1=fopen(*++argv,"r")) != NULL) && (argc == 3))  {
43:            fp2=fopen(*++argv,"wb");
44:            fputc(0x0d,fp2);
45:            fputc(0x0a,fp2);
46:            while((c = getc(fp1)) != EOF)   {
47:            if (co==16){
48:                    fputc(0x0d,fp2);
49:                    fputc(0x0a,fp2);
50:                    co=0;
```

```
 51:                    }
 52:                    switch (c)       {
 53:                    case '&':
 54:                            if(sw == 1)      {
 55:                                    fputs(b_hexS(buf,128-(sum&127)),fp2);
 56:                                    fputc(' ',fp2);
 57:                                    co=co+1;
 58:                                    sw = 0;
 59:                            } else {
 60:                                    sw = 1;
 61:                                    sum = 0;
 62:                            }
 63:                            break;
 64:                    case '$':
 65:                            j = 16;
 66:                            break;
 67:                    case '#':
 68:                            j = 10;
 69:                            break;
 70:                    case '%':
 71:                            j = 2;
 72:                            break;
 73:                    case 'a':       case 'b':       case 'c':
 74:                    case 'd':       case 'e':       case 'f':
 75:                            c = c&0xdf;
 76:                    case 'A':       case 'B':       case 'C':
 77:                    case 'D':       case 'E':       case 'F':
 78:                            if (j == 10) {
 79:                                    break;
 80:                            }
 81:                            c = c-7;
 82:                    case '2':       case '3':       case '4':
 83:                    case '5':       case '6':       case '7':
 84:                    case '8':       case '9':
 85:                            if (j == 2) {
 86:                                    break;
 87:                            }
 88:                    case '0':       case '1':
 89:                            if (s==1)       {
 90:                                    i = i*j+c-'0';
 91:                            } else {
 92:                                    i = c-'0';
 93:                                    s = 1;
 94:                            }
 95:                            break;
 96:                    case '*':
 97:                            while((c = getc(fp1)) != '¥n')  {
 98:                            }
 99:                    default:
100:                            if (s == 1)     {
101:                            if (i<16){fputc('0',fp2);}
102:                                    fputs(b_hexS(buf,i&0xff),fp2);
103:                                    fputc(' ',fp2);
104:                                    co=co+1;
105:                                    if (sw == 1)    {
106:                                            sum=sum+i;
107:                                    }
108:                                    s = 0;
109:                            }
110:                            break;
111:                    }
112:            }
113:            if(sw == 1)     {
114:                    fputs(b_hexS(buf,128-(sum&127)),fp2);
115:                    fputc(' ',fp2);
116:                    co=co+1;
117:            }
118:            fclose(fp1);
119:            fclose(fp2);
120:    } else {
121:            puts(" Usage: mk_exc 'source text file' 'exclusive data file'.")
;
122:    }
123: }
```

## リスト2

```
*
*        MK_EXC.C のサンプル・データです。
*        送り込みには MIDI.EXE などを使って下さいね。
*
*        なお、このデータはMT－32 Old Versionには対応していません。
*        ちなみにOld VersionというのはPHONE端子が付いてない奴のことです。
*        あと、音色を定義しているのでその点も注意して下さい。
*

*
*                CM64/32L,MT-32 Software Reset data.
*                Written Oct. 17, 1990 by Tama-chan.
*
*                このメッセージでMT／CMシリーズは初期化される.
*
*
*f0
*41                                  * Maker ID. Roland
*10                                  * Dvice ID. 1ch.
*16                                  * Model ID. CM-64/32L,MT-32,D-10/20/110
*12                                  * data set 1
*&                                   * check sum start
*7f 00 00                            * All Parameters Reset
*00
*&                                   * check sum end
*f7

*
*                CM64/32L,MT-32 System setup data.
*                Written May 16, 1990 by Tama-chan.
*
F0
41                                   * Maker ID. Roland
10                                   * Dvice ID. 1ch.
16                                   * Model ID. CM-64/32L,MT-32,D-10/20/110
12                                   * data set 1
&                                    * check sum start
10 00 01                             * System area. Reverb mode.
01                                   * Reverb mode Holl
04                                   * Reverb Time 5
04                                   * Reverb Lebel 5
06 06 06 04 04 00 02 00 04           * Partial Reserve
01 02 03 04 05 06 07 08 09           * MIDI Channel
64                                   * Master Volume
&                                    * check sum end
F7


*
*                CM64/32L,MT-32 Tone data.
*                Written May 16, 1990 by Tama-chan.
*

F0
41                                   * Maker ID. Roland
10                                   * Dvice ID. 1ch.
16                                   * Model ID. CM-64/32L,MT-32,D-10/20/110
12                                   * data set 1
&                                    * check sum start
08 00 00                             * Timbre &1
                                     * Common parameter
54 69 6D 70 61 6E 69 4C 6F 77        * TimpaniLow
03 00 03 00

11 34 03 01 00 2E 00 07              * WG
00 00 00 00 00 00 00 32 32 32 32 32  * Pitch ENV
00 00 00                             * LFO
00 00 03 00 07                       * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00 * TVF ENV
64 41 48 0C 08 0C                    * TVA
00 04 00 00 00 00 46 64 64 64 64     * TVA ENV

11 30 03 01 00 01 00 07              * WG
00 00 00 00 00 00 00 32 32 32 32 32  * Pitch ENV
00 00 00                             * LFO
00 00 03 00 07                       * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00 * TVF ENV
64 41 48 0C 08 0C                    * TVA
00 04 00 00 00 00 46 64 64 64 64     * TVA ENV

11 30 03 01 00 2E 00 07              * WG
00 00 00 00 00 00 00 32 32 32 32 32  * Pitch ENV
00 00 00                             * LFO
00 00 03 00 07                       * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00 * TVF ENV
64 41 48 0C 08 0C                    * TVA
00 04 00 00 00 00 46 64 64 64 64     * TVA ENV

24 32 0B 00 00 00 00 07              * WG
00 00 00 00 00 00 00 32 32 32 32 32  * Pitch ENV
00 00 00                             * LFO
00 00 0B 00 07                       * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00 * TVF ENV
00 32 5B 0C 1B 0C                    * TVA
00 00 00 00 00 00 00 00 00 00        * TVA ENV
&                                    * Check sum end
F7

F0
41                                   * Maker ID. Roland
10                                   * Dvice ID. 1ch.
16                                   * Model ID. CM-64/32L,MT-32,D-10/20/110
12                                   * data set 1
&                                    * check sum start
08 02 00                             * Timbre &2
                                     * Common parameter
54 69 6D 70 61 6E 69 48 69 20        * TimpaniHi
03 00 03 00

18 34 03 01 00 2E 00 07              * WG
00 00 00 00 00 00 00 32 32 32 32 32  * Pitch ENV
00 00 00                             * LFO
00 00 03 00 07                       * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00 * TVF ENV
64 41 48 0C 08 0C                    * TVA
00 04 00 00 00 00 46 64 64 64 64     * TVA ENV

18 30 03 01 00 01 00 07              * WG
00 00 00 00 00 00 00 32 32 32 32 32  * Pitch ENV
00 00 00                             * LFO
00 00 03 00 07                       * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00 * TVF ENV
64 41 48 0C 08 0C                    * TVA
00 04 00 00 00 00 46 64 64 64 64     * TVA ENV

18 30 03 01 00 2E 00 07              * WG
00 00 00 00 00 00 00 32 32 32 32 32  * Pitch ENV
00 00 00                             * LFO
00 00 03 00 07                       * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00 * TVF ENV
64 41 48 0C 08 0C                    * TVA
00 04 00 00 00 00 46 64 64 64 64     * TVA ENV

24 32 0B 00 00 00 00 07              * WG
00 00 00 00 00 00 00 32 32 32 32 32  * Pitch ENV
00 00 00                             * LFO
00 00 0B 00 07                       * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00 * TVF ENV
00 32 5B 0C 1B 0C                    * TVA
00 00 00 00 00 00 00 00 00 00        * TVA ENV
&                                    * Check sum end
F7

F0
41                                   * Maker ID. Roland
10                                   * Dvice ID. 1ch.
16                                   * Model ID. CM-64/32L,MT-32,D-10/20/110
12                                   * data set 1
&                                    * check sum start
08 04 00                             * Timbre &3
                                     * Common parameter
45 2E 54 6F 6D 20 4D 69 64 20        * E.Tom Mid
05 00 01 01

1d 32 03 01 00 38 00 07              * WG
05 03 02 11 17 42 64 37 2A 1F 11 01  * Pitch ENV
00 00 00                             * LFO
00 00 0B 00 07                       * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00 * TVF ENV
63 4E 7F 0C 00 0C                    * TVA
00 04 00 64 64 64 64 64 64 64 00     * TVA ENV

1d 32 03 01 00 38 00 07              * WG
05 03 02 11 17 42 64 37 2A 1F 11 01  * Pitch ENV
00 00 00                             * LFO
00 00 0B 00 07                       * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00 * TVF ENV
63 4E 7F 0C 00 0C                    * TVA
00 04 00 64 64 64 64 64 64 64 00     * TVA ENV

1d 32 03 01 00 38 00 07              * WG
05 03 02 11 17 42 64 37 2A 1F 11 01  * Pitch ENV
00 00 00                             * LFO
00 00 0B 00 07                       * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00 * TVF ENV
63 4E 7F 0C 00 0C                    * TVA
00 04 00 64 64 64 64 64 64 64 00     * TVA ENV

1d 32 03 01 00 38 00 07              * WG
05 03 02 11 17 42 64 37 2A 1F 11 01  * Pitch ENV
00 00 00                             * LFO
00 00 0B 00 07                       * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00 * TVF ENV
63 4E 7F 0C 00 0C                    * TVA
```

```
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV
&                                           * check sum end
F7

F0
41                                          * Maker ID. Roland
10                                          * Dvice ID. 1ch.
16                                          * Model ID. CM-64/32L,MT-32,D-10/20/110
12                                          * data set 1
&                                           * check sum start
08 06 00                                    * Timbre &4
                                            * Common parameter
45 2E 54 6F 6D 20 4D 69 64 20               * E.Tom Mid
05 00 01 01

1d 32 03 01 00 38 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV

1d 32 03 01 00 38 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV

1d 32 03 01 00 38 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV

1d 32 03 01 00 38 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV
&                                           * check sum end
F7

F0
41                                          * Maker ID. Roland
10                                          * Dvice ID. 1ch.
16                                          * Model ID. CM-64/32L,MT-32,D-10/20/110
12                                          * data set 1
&                                           * check sum start
08 08 00                                    * Timbre &5
                                            * Common parameter
45 2E 54 6F 6D 20 48 69 67 68               * E.Tom High
05 00 01 01

29 32 03 01 00 38 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV

29 32 03 01 00 38 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV

29 32 03 01 00 38 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV

29 32 03 01 00 38 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV
&                                           * check sum end
F7

F0
41                                          * Maker ID. Roland
10                                          * Dvice ID. 1ch.
16                                          * Model ID. CM-64/32L,MT-32,D-10/20/110
12                                          * data set 1
&                                           * check sum start
08 0A 00                                    * Timbre &6
                                            * Common parameter
42 61 20 20 20 20 20 20 20 20               * Bass Drum
05 00 01 00

1D 32 03 01 00 00 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV

1D 32 03 01 00 36 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV

1D 32 03 01 00 00 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV

18 32 03 01 00 38 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
```

```
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV
&                                           * check sum end
F7

F0
41                                          * Maker ID. Roland
10                                          * Dvice ID. 1ch.
16                                          * Model ID. CM-64/32L,MT-32,D-10/20/110
12                                          * data set 1
&                                           * check sum start
08 0C 00                                    * Timbre &8
                                            * Common parameter
45 2E 54 6F 6D 20 4C 6F 77 20               * Snear
05 00 01 00

1D 32 03 01 00 01 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV

18 32 03 01 00 38 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV

18 32 03 01 00 38 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV

18 32 03 01 00 38 00 07                     * WG
05 03 02 11 17 42 64 37 2A 1F 11 01         * Pitch ENV
00 00 00                                    * LFO
00 00 0B 00 07                              * TVF
00 00 00 00 00 00 00 00 00 00 00 00 00      * TVF ENV
63 4E 7F 0C 00 0C                           * TVA
00 04 00 64 64 64 64 64 64 64 00            * TVA ENV
&                                           * check sum end
F7


*
*
*               CM64/32L,MT-32 Drums part assign data.
*               Written May 16, 1990 by Tama-chan.
*
*               YAMAHA SY22のDrums kidのアサインに準拠.
*
F0
41              * Maker ID. Roland
10              * Dvice ID. 1ch.
16              * Model ID. CM-64/32L,MT-32,D-10/20/110
12              * data set 1
&               * check sum start
03 01 40        * data address. Rhythm part setup area. note No. A1.
                * note No.      Tone name             No.,Vol,LR,Rev

05 64 07 01     * note C2       Acoustic Bass Drum    r01,100,><,on
7F 00 07 00     * note C&2                            r64,000,><,off
06 64 07 01     * note D2       Acoustic Snare Drum   r02,100,><,on
7F 00 07 00     * note D&2                            r64,000,><,off
02 64 03 01     * note E2       E.Tom Low             i03,100,4<,on
03 64 0B 01     * note F2       E.Tom Mid             i04,100,>4,on
45 64 09 01     * note F&2      Elec Snear            r06,100,>2,on
04 64 05 01     * note G2       E.Tom Hi              i05,100,2<,on
05 64 07 01     * note G&2      Acoustic Bass Drum    r01,100,><,on
05 64 07 01     * note A2       Acoustic Bass Drum    r01,100,><,on
4A 64 06 01     * note A&2      Rim Shot              r11,100,1<,on
44 64 0B 01     * note B2       Acoustic Low Tom      r05,100,>5,on

43 64 08 01     * note C3       Acoustic Middle Tom   r04,100,>1,on
06 64 07 01     * note C&3      Acoustic Snare Drum   r02,100,><,on
43 64 08 01     * note D3       Acoustic Middle Tom   r04,100,>1,on
4A 64 06 01     * note D&3      Rim Shot              r11,100,1<,on
06 64 07 01     * note E3       Acoustic Snare Drum   r02,100,><,on
42 64 03 01     * note F3       Acoustic High Tom     r03,100,4<,on
4B 64 08 01     * note F&3      Hand Clap             r12,100,>1,on
4C 64 07 01     * note G3       Cowbell               r13,100,><,on
5B 64 09 01     * note G&3      Cabasa                r28,100,>2,on
46 64 06 01     * note A3       Closed High Hat       r07,100,1<,on
48 64 06 01     * note A&3      Crash Cymbal          r09,100,1<,on
47 64 06 01     * note B3       Open High Hat 1       r08,100,1<,on

48 64 06 01     * note C4       Crash Cymbal          r09,100,1<,on
7F 00 07 00     * note C&4                            r64,000,><,off
7F 00 07 00     * note D4                             r64,000,><,off
49 64 08 01     * note D&4      Ride Cymbal           r10,100,>1,on
4F 64 0A 01     * note E4       Low Conga             r16,100,>3,on
4E 64 09 01     * note F4       High Conga            r15,100,>2,on
4D 64 08 01     * note F&4      Mute High Conga       r14,100,>1,on
7F 00 07 00     * note G4                             r64,000,><,off
7F 00 07 00     * note G&4                            r64,000,><,off
51 64 05 01     * note A4       Low Timbale           r18,100,2<,on
50 64 07 01     * note A&4      High Timbale          r17,100,><,on
56 64 09 01     * note B4       Tambourine            r23,100,>2,on

4B 64 08 01     * note C5       Hand Clap             r12,100,>1,on
57 64 0C 01     * note C&5      Claves                r24,100,>5,on
55 64 02 01     * note D5       Low Agogo             r22,100,5<,on
54 64 02 01     * note D&5      High Agogo            r21,100,5<,on
7F 00 07 00     * note E5                             r64,000,><,off
7F 00 07 00     * note F5                             r64,000,><,off
59 64 09 01     * note F&5      Long Whistle          r26,100,>2,on
5A 64 09 01     * note G5       Short Whistle         r27,100,>2,on
7F 00 07 00     * note G&5                            r64,000,><,off
7F 00 07 00     * note A5                             r64,000,><,off
4C 64 07 01     * note A&5      Cowbell               r13,100,><,on
7F 00 07 00     * note B5                             r64,000,><,off

7F 00 07 00     * note C6                             r64,000,><,off
7F 00 07 00     * note C&6                            r64,000,><,off
7F 00 07 00     * note D6                             r64,000,><,off
7F 00 07 00     * note D&6                            r64,000,><,off
7F 00 07 00     * note E6                             r64,000,><,off
7F 00 07 00     * note F6                             r64,000,><,off
7F 00 07 00     * note F&6                            r64,000,><,off
7F 00 07 00     * note G6                             r64,000,><,off
7F 00 07 00     * note G&6                            r64,000,><,off
00 64 07 01     * note A6       TimpaniL              i01,100,><,on
7F 00 07 00     * note A&6                            r64,000,><,off
01 64 07 01     * note B6       TimpaniH              i02,100,><,on

&               * check sum end
F7
```

Creative Computer Music入門(4)

# 楽曲の基本，終止形

Taki Yasushi 瀧　康史

**今回は，代理コード，終止形，借用和音の3つについて解説していきたいと思います。特に，終止形は楽曲においての基本ですので，しっかりマスターしてください。できれば自分で4～8小節程度の曲を作ってみるといいでしょう。**

## バッハは小川ではない大海だ!!

バッハ，なにはともあれバッハ！　バッハといっても小川ではありません。大海です！　すべての川は大海に流れるのです。上の小見出しは，あのベートーベンのお言葉。口数少ない彼の名台詞です。

バッハを語らせたらきりがない人って，いっぱいいます。逆にバッハをあんまりよく評価しない人もいます。「基礎はバッハで完成し，ほかの音楽はそれに個人的な趣味を継ぎ足しただけだ」というほどバッハに惚れ込む人さえいます。それだけバッハはよいのでしょうか？ 問答無用よいのです。も，ぐ！（も～うGood！　という意味）

要約すると毎度のごとく，バッハのCDを買っちゃったらそれにはまっちゃったのよ，といってただけだったりして。やっぱりバッハはいつ聴いてもいいのです。なんといっても，厳かな気分になれる自分が最高。といっても，今日はこのCDの中身には触れません。だって，一緒についている小冊子から引用したみたいになりますから。

というわけで，バッハのコメントはパスします。詳しく知りたい人は，CDを買ってみましょう。損はしないと思いますよ。本屋さんで売ってます。TCD-C15というコードネームです。うん。これだけわかればすぐ見つかるでしょう。

## 代理コード

今回は，スリーコードの発展について考えてみましょう。スリーコードというのは，いわずと知れたトニック，ドミナント，サブドミナントの3つです。前回の話の中に，「スリーコードの名曲というのが……」っていうのがありました。もちろん，スリーコードのみでかなり曲は作れますが，単調になる可能性がかなりありえます。そこでスリーコード以外のコードを使うにはどのようにすればいいか？　どのように曲のイメージを広げていくかに今回は重点をおいて，進めていこうと思います。

さて，小見出しには代理コードと書きましたが，まずはダイアトニックコードのおさらいから進めていきましょうか。ダイアトニックコードは，連載1回目の最後でお話ししました。図1を見てください。1回目ではマイナーが小文字（i iv v)だとか，オーギュメントが＋とか，ややこしい記号を使いましたが，今回からはわかりやすいように，記号はスケール上の度数をローマ数字で書いてそれにmとかaugとかをつけて説明したいと思います。え？　いってる意味がわからない？　要するに，IIm, VIIaugなどというように書くということです。

さて，メジャースケール上のダイアトニックコードは，図1に書いてあるように，I,IIm,IIIm,IV,V,VIm,VIIdimになります。詳しいことは，1回目の最後と2回目の前回の復習のコラムを見てください。このうち，I度はトニック，IV度はサブドミナント，V度はドミナントになります。

それぞれ，

**●トニック（T）**：もっとも安定したコード。どんなコード進行でもこのコードに進むことにより落ち着きます。

**●サブドミナント（S）**：不安定なコード。次のコードに進みたがるコードです。

**●ドミナント（D）**：トニックに進みたがるコード。これは7th，すなわちここでは短7度の音を加えて，V7(ドミナントセブンス)として使われる場合がほとんどです。と，まあもっとも重要なコードたちです。これらを前知識として代理コードに進んでみましょう。

スリーコード以外のダイアトニックコードを見てみましょう。図1をよく見ると，これらのトライアドコード（3音で構成されたコード）のうち，スリーコードと似ているもの，すなわち2つまでが同じものが結構あります。これらのコードはスリーコードの代わりに使うことができ，これらを，「代理する」すなわち，代理コードというわけです。それぞれの表を表1に書いておきますから，参考にしてください。

IIImは，第1転回形ではドミナントに近い形を示すので，とりあえずドミナントにも入れています。気をつけなくてはならないことは，代理コードは転回形を好まないということ。いずれの場合も，転回は第1段階まで，第2からは（マイナースケールのIIm, IIIを抜く）**代理としての機能が失われて**しまいます。理由はいまは，そっとおいておきましょう（いずれ話すつもりです)。

代理コードってわかりましたか？

## カデンツってなに？

さて，前回はとっても基本的なコード進行に触れてみました。説明したコード進行の種類は表2に書いておきます。これらの

表1 代理コード表

| 機　能 | スリーコード | 代理コード |
|---|---|---|
| トニック | I | IIIm　VIm |
| サブドミナント | IV | IIm |
| ドミナント | V | VIIdim　III$^{\wedge 1}$ m |

図1 スリーコード以外のダイアトニックコード

コード進行は，スリーコードの機能から簡単に割り出せるものです。これらはI（トニック）から始まり，必ずIに戻ってくるので，この3種類は循環コードと呼ばれます。

もちろん，すべての楽曲のなかで当てはまるわけではないですけど，基本中の基本です。しっかり身につけておいてください。この，循環コードをこれから「カデンツ」と呼びましょう。カデンツア，ケーデンスともいいます。終止形というのもこのことです。

実のところ，カデンツ（終止形）には，全終止(完全終止，正格終止)，半終止，偽終止，変終止の4つがあるのですが，これらを合成すると，つまるところ図2のような形に（基本的には）なるというところで妥協してください。うっっ，苦しい，なぜ，完全に説明しないかといいますと，全終止には6つの変形が，半終止には9つ。偽終止には3つと，合わせてしまうとかなりの数があるのです。それに，私がいったスケールの種類は2つ，マイナーとメジャーしかないといった嘘が，もろに現れていて，ナポリ形やら，ドリア形やら，5度7下変○転回などという，わけわかめな専門用語を説明しなくてはならないので，ここは基本的なものだけ説明したいと思います。ただ，一般に使うときは表2のカデンツを利用するとします。表2を覚えていればたいていのことはわかりますからね。

また，この3つだけでは単純であっても，さっき説明した代理コードをこの中におり交ぜたり，カデンツの基本的な変化をすることによって，多彩なコード進行を生み出してくれます。代理コードは図1と表1をよく見ればわかるとおり，導き出すことができますが，あえてすべて覚える必要もないでしょう。その都度，導き出せばいずれ身についてくると思います。

とりあえず，基本的なカデンツについて紹介しましょう。

**表2 カデンツの短縮表**

| 1 | I－IV－I | C－F－C |
|---|---|---|
| 2 | I－V－I | C－G(7)－C |
| 3 | I－IV－V－I | C－F－G(7)－C |

**○全終止(完全終止，正格終止)**

V(7)－I（D－T）の進行です。ドミナントの機能そのままの簡単なものです。曲の段落のうちもっとも強力なもので，楽曲の最終的な終止，またはそれに準ずる大きな段落に用いられます。

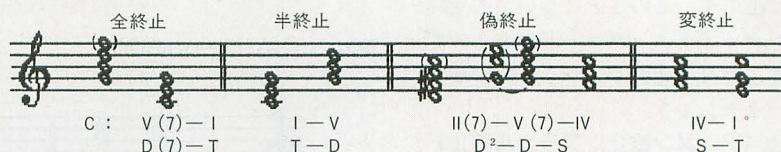
全終止のなかでも特にこの形は音のエネルギーの減衰量から見て，「解決」といわれます。

**○半終止**

I－V（T－D）の進行です。終止形のうちもっとも不安定なもので，次のフレーズへの進行の期待を持って終了します。Vの基本形（転回していないもの）がもっともふさわしく，7thや転回形は使用しないほうが無難です。その理由は，やってみればわかるのですが，段落が終わったという感じが弱いのです。また，Vの代わりにIVが使われる場合もあります。

全終止とおり交ぜて使う場合がほとんどですね。

**○偽終止　　タイ**

**図2 終止形の基本**

## 前回の復習とうんちく

前回，初めて簡単な「アレンジ」に取り掛かってみました。ただの伴奏つけだけですけど，カンがいい人，というか意欲的な人は，もう自分自身で作曲していたり，アレンジしてみたりしてるのかもしれませんね。チャレンジ精神が旺盛だとこっちもやりやすいです。試行錯誤でいろいろやってみてください。では，やってない人も，やってる人も含めて前回のおさらいをしてみましょう。

前回の主旨のひとつは，終止形の基本でした。簡単にはずれない音は何か？ と説明したわけですけど，法則は当然のごとくまだまだあったりします。ただ，私自身法則だけにこだわる主義ではありませんし，「アレンジができるようになる」のが連載通じての主旨であって，「法則を覚える」のは二の次ですから，難しいことはパスします(面倒臭いことはあとあとにね)。あっと，話がずれてしまいましたね。

もうひとつの主旨は，アレンジ（それも伴奏に限って）のポイントを考え，それについての意味を述べてみました。私が考えた伴奏をつけるにあたってのポイントは，

1．メロディを殺さない
2．ハーモニーをくずさない
3．それでいて曲に厚みやメリハリを持たせる

の3つです。

1については，直感的にわかると思います。メロディを殺さない。すなわち，メロディを生かすということを考えたわけです。

たとえば，一般的な耳の肥えていない人を想定しましょう。シンガーの歌う歌を聴くとき，もちろん彼はシンガーの声，すなわちメロディを聞いているでしょう。よほどのことがない限り，バックのリズムの刻みや，音のハーモニーまでは（たとえ聴こえていたとしても）しっかりは聴いていないと思います。仮に，聴いているとしたら，オブリガッドぐらいなものでしょう。

このようにほとんどの人は，メロディを重点的に聴いている場合がほとんどです。ある特定の人を狙った曲を作るならともかく（たとえばドラマー向けだとか），一般受けをする曲，すなわち，いろんな人が「いいな，この曲」っていってくれる曲を狙って作る，またはアレンジするには，メロディを殺すか生かすはその曲を左右する死活問題といって間違いないでしょう。メロディを生かすというのはどういうことか？これは特にアレンジには問題になってきます。

主義としてあげるには簡単な問題ですが，実はこれが奥が深いものです。なぜなら，たいてい曲を作る人と聴く人は違う耳で聴いているからです。アレンジというからには，原曲があるわけですから，どのメロディラインを生かすかどうかは，アレンジの筋を決める大変な問題となるでしょう。

2の「ハーモニーをくずさない」について。音を加えるとき，特に長目に発音する音は特に注意して扱う必要があります。たとえば，拍子の分母。つまり，$\frac{6}{8}$なら，8分音符よりも長く発音する音には，より注意する必要があるということです。これらの音は，まず，コードの構成音以外は使わないほうが無難でしょう。少なくとも，いまの段階では使わないほうが無難です。このことについては，今回やることについてかなりの結びつきがあります。

3の「厚みやメリハリを持たせる」ですが，これは，アレンジの醍醐味でもあります。あ，もちろん，明らかに一度にたくさんの楽器を使っているオーケストラのような演奏を，FM音源8音で演奏できるように音をいくらかはしょるアレンジもありますが，基本的にはより多くの音を足してゴージャスにするアレンジのほうが，断然面白いと（少なくても私は）思います。

ゴージャスにするにはどうすればいいか。音をたくさん入れる。いろんな成分を含む音を入れる。などという，ごくありふれたことです。ただ，これが難しく，どんな音ははずれないか，どんな音がメリハリを持たせるかは，臨機応変でなかなか簡単に導き出せるものではありません。

やっぱり，よい音楽を聴くことでしょうね。基本は。

さてもうひとつは，サンプルの伴奏パターンを紹介したわけですが，これはもういいですよね？ いまさら，これに対してもう一度説明するわけにはいかないので，この点に対しては，12月号をご覧ください。

II(7)－V(7)－IVの進行です。これがなぜ偽終止と呼ばれたかといいますと，全終止のIがIVに変化しているからなのです。実際には，かなり誇張されたクライマックスを作るので，使い方にはかなり注意が必要です。たとえば，

イ）あまり短い曲には不向き。短い曲に大袈裟なクライマックスがくると，興ざめでしょ。

ロ）何度も使わない。使いすぎるとクライマックスにはならないですからね。

ハ）終わりに近い部分で使うこと。クライマックスが早すぎるとおかしいからね。

ニ）必ずそのあと，全終止か変終止で尻拭いをすること。

進行としては，

II—V(7) < I(全終止) / VI(偽終止) < 全終止 / 偽終止

という進行が「ぐ(Good)」でしょう。

いちばん最初にあるIIですが，これはV度調のV度です。このことについての意味は，ノンダイアトニックコードのところで話します。今月の範囲ですから，気にせず進んでいきましょう。

**○変終止（アーメン終止，プラガル終止）**

全終止や偽終止のあとに，IV(IVm, IVm6, IVdim)－Iと使われることが多い。変終止は，全終止と同じくかなり安定した終止感を持っています。

IV－Iというのが変終止の基本的な形ではありますが，実際にはいろいろな形に変化したり，組み合わされたりして使われています。

## ノンダイアトニックコード(借用和音)

ここまでお話ししたコード進行は，基本的には同一スケール上にできるコードから導き出されたものでした。これだけだと，単調さを招きやすいし，曲が一本調子になりやすくなってしまいます。

そこで，広い調性感をえるために，ダイアトニックコード以外のを使う方法があります。前回，スケール以外の音ははずれやすい（ようなこと）といったのだけど，それを裏切ってしまうような方法です。

ノンダイアトニックコード，すなわち借用和音の持つ役割は，一時的にほかの調の和音を用いることによって，和声に色彩的あるいは機能的変化を与えることにあります。たとえば，半終止のVで終わるものを強調するために，その直前にV調，すなわちC調のスケールなら，一時的にG（V）のVを（これをvV(7)と表しましょう）おき，そのあとのVをはっきりさせるためにあるわけです。これがなぜうまくいくかといいますと，vVにはダイアトニックコードにない，特徴となる音が含まれているため（特徴音）なのです（図3）。

もうひとつの例を挙げると，たとえば，I－VIの進行。この2つの和音は，どちらとも，トニックとしての機能を併せ持っていて，2つのコードの構成音も違いがひとつしかないために，あまり変化した感じがありません。そこで，VI調のVに1回進行してVIにいくと，通常の進行よりがぜんVIの存在がはっきりしてくるわけです。進行は，

I－viV(7)－VIになるわけですね（図4）。

あと，さっきの偽終止のところでちょっといっていたのもまさにこれです。最初のIIは実はV度調のV，すなわちvV(7)で一般にセカンダリードミナント（D²）と，特別にいわれています。つまるところ，簡単にいってしまえば，すべてのD7はII(7)－V(7)の進行に分解できるとでもいっておきましょうか(図5)。それくらいよくある進行なんですよ。

借用和音の使い方は，このように後ろを目立たせるための「手前の化粧」みたいな使い方のほかに，次のような使い方もあり

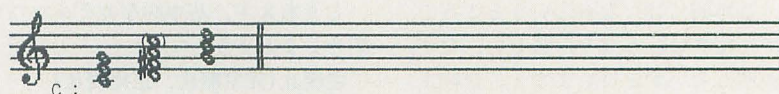
**図3 半終止に借用和音$_{\mathrm{V}}$V(7)を入れてみる**

I－$_{\mathrm{V}}$V(7)－V
*$_{\mathrm{V}}$V(7)－VはV調の$_{\mathrm{V}}$D（D²）－Tだから，あまりはっきりしない半終止も全終止のように落ち着く

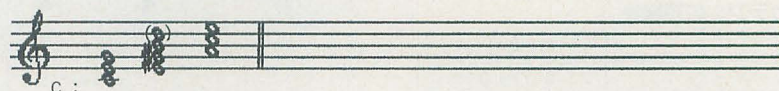
**図4 I－$_{\mathrm{VI}}$V(7)－VIの進行**

I－$_{\mathrm{VI}}$V(7)－VI
*I－VIに比べてメリハリが出る

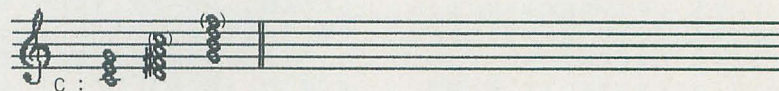
**図5 D7の分解**

I－II(7)－V(7)
T－D²(7)－D
*ポップスではわりとひんぱんに使われる

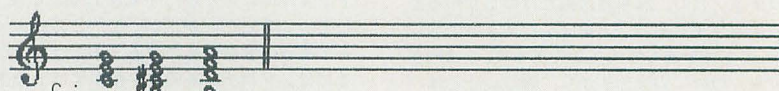
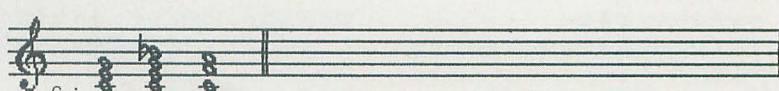
**図6 I→IIへの進行方法**

I－iiV(7)—IIm
(VI 7)
*通常ではI→IImは進めないがiiV(7)を挟めばスムーズに流れる

**図7 I→IVをより滑らかにする**

I—ivV7—IV
*ivV7を挟むとよりスムーズになる

ます。

たとえば，IとIIを連結するために，掛け橋として，iiV(7)を使ったり(図6)，IとIVをより滑らかに進行させるために間に，ivV(7)を挟んだり(図7)，借用和音の効果的な使い方はさまざまです。

このように，音楽の雰囲気を盛り上げ，その動きを潤滑にして，音の場を広げることに借用和音の意図があるのです。

これらの効果的な使い方は，なかなかセンスと知恵がうまく結びつかないとできないもので，これからの連載の鍵にもなるでしょう。今回1回で説明してしまうのは，カデンツ同様重すぎますので，あとあとちょっとずつ説明していきましょう。

## このメロディにこのコード!?

さて，今回はこのままでは和声の話ばかりになってしまうので，ちょっとぐらい厚みのない話をしましょう。メロディからコードを割り出す方法は次回にでもやってみるつもりですので，今回はさらっと読んでみてください。MMLの苦手な私が苦心して（たったのこれだけですけど）打ち込んだ代物だから，実際に打ってみて感覚的に聞いてほしいです，はい。

メロディからコードを割り出すことって意外と曲者なのですね。実に何とおりものコード進行に割り当てることができるってわけ。その代わり，コードを置き換えることによってその後の進行が変わってくるから，ま，意図的にいろいろできるっていえば面白いかな？　しかも，その気になれば同じメロでも，コードの割り付け方で悲しいメロにしたり明るいメロにしたりいろいろできるのです。すごいでしょ？　つまるところ，コードって1カ所に何種類も割り当てられるんですよ。難しいかなあ。そうだよね，これだけじゃわけわかめだよね？　では感性に訴えてみましょう。図8を見てください。

はい，たった4小節のメロです。メロは簡単。楽器がない人でもすぐにイメージが伝わるように，FM音源で鳴らすことができるzmsファイルをつけてみました。

図8のメロにはA（リスト1）とB（リスト2）の2種類のコードを割り当ててあります。どっちもそれはそれで合うはずです。zmsファイルを演奏するなり，自分で鍵盤で弾くなりしてください。どうです？　雰囲気が違うでしょ？　それなのにメロは一緒。単調なメロなら，いかにコードで操作できるかがわかっていただけたでしょうか。それに，AとBではこのあとに思いつくメロディの雰囲気はちょっと変わっていくでしょう？　この場合，伴奏についてはよりわかりやすくするため，コード進行のみにさせてもらいました。これにどんな伴奏が合うかは，各自先月号を引っ張り出して研究してください。

ほかにはこんなのもあります。図9を見てください。一応zmsファイルを入れておきます(リスト3)。どうでしょう？　曲風は？　なかなか悲しい感じでしょ？　でも，マイナーではなくて，メジャーの王道を走ってます。ところがメロが結構悲しくて強烈な印象があるため，いくらコード進行がメジャーでも，暗いイメージ，物悲しいイメージは出来上がってしまうわけです。ちなみにこの曲はこのあとも，メジャーのコード進行で突っ走ります。

ところで，私は何をいっているのでしょう？　私が採譜した曲のコードが本物と違ったときに逃れるため？　それもあるんですけど，とりあえずここではそうじゃなくて一見雰囲気が固定的な曲でも，コードの割り付け次第で，曲想がずいぶん変わりま

**図8 2種類のコード進行をあてはめる**

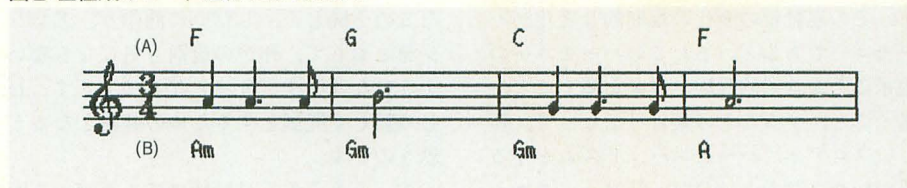



**図9 サンプル**

**リスト1**

```
(i)
(aFM1,1)(m1,1000)
(aFM2,2)(m2,1000)

/           F        G        C        F

(t1) @1o4  a4a4.a8  b2.      g4g4.g8  a2.
(t2) @1o3  'c2.fa'  'd2.gb'  'c2.eg'  'c2.fa'

(p)
```

**リスト2**

```
(i)
(aFM1,1)(m1,1000)
(aFM2,2)(m2,1000)

/           Am       Gm       Gm       Am

(t1) @1o4  a4a4.a8  b2.      g4g4.g8  a2.
(t2) @1o3  'c2.ea'  'd2.gb'  'd2.gb'   'c2.ea'

(p)
```

**リスト3**

```
(i)

(aFM1,1)(m1,5000)
(aFM3,2)(m2,5000)

/       Title

.comment Simphony  "Like Your Eternal Lover." Segment 2nd から。

(o52)   /       Lento

/               C

(t1) @1 o4 l8 <'c.e'>g16g4.g<'e16.c''d16.>b''c16>a'
(t2) @1 o3 l8 @d1>cg<e>g<e>g<e>g@d0

/    G

(t1) 'd.>b'>'b16g''d4.g''dg'<'d16.>b''c16.>a'>'b16g'
(t2) @d1>b<g<d>g<d>g<d>g@d0

/    F

(t1) <'c.>a'>'a16f''f4.c''f8c'<'c16.>a'>'b16.g''a16f'
(t2) @d1>a<f<c>f<c>f<c>f@d0

/    E

(t1) 'g#2e''g#.e''a.f''bg'
(t2) @d1>b<g#<e>g#<e>g#<e>g#@d0

(p)
```

すよ，といいたかったのです。場合によってはちょっと雰囲気を変えた粋なアレンジだってできることにもなりますよね？ たとえば，図９なんかは左手の伴奏パターンで曲想が変わります。面倒臭いので例は示しませんが。

さて，ここで余談。アレンジの講座も今回で４回目となるわけですが，前回でちょっと違った講座をしただけで，ほとんど和声の話ばかりでした。和声すなわちハーモニーはそれほどまでに必要なことなのでしょうか？ たとえば難しい言葉を並べてしまえば，楽曲とは音の運動の集大成であって，その運動量の変化の規則的な変化がハーモニーであると。またこのハーモニーの追求こそが音楽的経験，音楽を聴くこと，書くこと，することの裏付けとなって，新しいメロディ，ハーモニー，リズムを作る手がかりになると。ただ，私はハーモニーに対する数々の約束事は，常識であって拘束では決してないと思うのです。連載の１回目の冒頭でいっていた先人たちの偉大な知識とはまさにそれで，新しい音楽シーンを築くためには，これを踏み台にしなくてはならないと思うからです。「お決まり」はしだいに身についていくもので，身についたらうまい使い方，「コツ」を覚えていきます。そして「お決まり」はもはや，足かせではなくなります。それは，車の運転と似ていて，初めてのときはあんなに面倒に感じた「車の操り方」がしだいに身につき（車に乗れない人，よい例がなくてごめんなさい），いずれは感性のまま運転できるようになるのと同じく，しだいに曲作り，アレンジをとおして，和声の規則も気にならないようになります。きっとなるはずです。むしろ新しく発展をするための基礎となると思うのです。

ま，そろそろくじけ始めてる人がいる頃だと思うので，ちょっと励ましてみただけです。このあたりから難しくなりますので，頑張ってくださいね。

## おわりに

さて，今回やったことを整理してみましょう。最初にやったのは代理コード。まあ，この代理コードの概念はとっかかりが比較的楽だと思うので，なぜ代理するのか？ とか，なぜ代理しても変ではないか？ など，追求しない限り簡単に身につくでしょう。どうしてもそういうものを追求したい人は，とりあえずこれはこんなものだと思っていて，あとで和声を極めていったうえで追求してみてください。その頃はきっと追求しなくても自然にわかってしまうでしょう。

次にカデンツ。これからいろんな曲をサンプルにしていきますが，どの曲もカデンツで表しながら進めていくつもりですので，頑張って理解してください。今回のメイン

## 音程（インターバル）

第１回目で「全然わからないや」と詰まってる人の天敵はやっぱりこれでしょう。本当は「どのようなことがわからなかったですか？」とおハガキを求めるのが情だと思うのですが，ここは独断と偏見で私が選ばせていただきました（もちろんそういったおハガキも大歓迎ですよ）。名付けて，「ズバリ!! 私はここがわからない!!」のコーナーです（勝手にコーナー作るなよ……）。

本題に移りましょう。まず音程それ自身はわかりますよね？ そうそう，前いったとおり，２つの音の連なりをいいます。でも，今回ここでいう音程はむしろそれではなくて，「音程差」といったほうが正しいでしょう。てっとり早くいってしまえば，２つの音の間隔のことをここでは指しています。

ここで問題になるには，その間隔を表すための単位です。たとえば，長さはcmで表しますし，エネルギーの大きさはW（ワット）やJ（ジュール）などが使われますよね？ 音程（差）では何という単位を使うかというと，「〜度」と表します。

そうですね。１回目からさも暗黙の了解のように使われていた完全〜度，長〜度とかというあれです。あれはあれで一応のみ込んでいた人もいると思います。ただ，なぜ完全なのか，なぜ長なのか。それを知っている人は少ないと思うのです。そんな点を含めてこのコーナーでは説明したいと思います。

第２回復習のコラムを見るとわかるのですが，音程にはつぎの種類があります。全部で５種類です。

１．完全音程
２．長音程
３．短音程
４．増音程
５．減音程

この５種類には以下の関係があることを頭に入れてください。ただ，のぺ〜って聞いているより百倍わかるでしょう（誇張表現）。

１．減音程<>短音程<>長音程<>増音程
２．減音程<>完全音程<>増音程

注意：半音１つ分上がると右に１つ。１つ分下がると左に１つ移動します。１，２は別々に考えてくださいね。

では，これらのことを頭に入れたうえで，説明を始めましょう。この，「完全，長，短」を覚える前に，１度，２度という数え方を考えてみましょう。

楽譜による説明もしていますので，それを見ながら読んでください。

１度の音はわかってのとおり同じ音です。ま，ハモるのは当たり前でしょう。鍵盤上で見ても楽譜上で見てもCから見ればC（おんなじ鍵盤だよ，おんなじ線の上だよ）Dから見たらD。

２度の音はCから見てD，Dから見てE，そしてEから見たFです。ここでは間に挟む半音の数は数えないでくださいね。そうです。楽譜上では１つ上の音を指します。シャープ，フラットのことは考えないで。楽譜上で１つ上の音。鍵盤上でも隣の鍵盤。ここでは白鍵だろうから，Cの隣のD，Dの隣のE，Eの隣のFと覚えてください。

３度の音はCから見てE，Dから見てF，Eから見てG。楽譜上では２つ上の音です。そうですよね？ 並べるとちょうどよく上下に並ぶんですよね。もちろん，鍵盤上でも１個飛ばした隣の音だから，Cの隣の隣のE，Dの隣の隣のFです。

とんで，８度の音は１オクターブ上を表します。これは簡単にわかりますよね？

さてさて，だいたいわかりましたか？ ○度という数え方は単純にフラットも，シャープも考えず，楽譜上では上にいくつ分か（１度の音もあるから忘れないでね）鍵盤上ならいくつ隣か？ そうやって覚えてください。実際使われる意味あいはもっと広いのですがこのコーナーの終わる頃にそれらのことをあかしましょう。いまは，これだけおさえてください。

さあ，整理しましょう。このなかで，完全とつくのはいくつありますか？ 長，短とつくのはいくつありますか？ 答えは，完全とつくのは，１度，４度，５度，８度の４つです。それから長，短がつくのは残りの２度，３度，６度，７度です。

ここで第１回で説明した倍音の話を思い出してください。とりあえず，ひと言で説明しちゃうと，共鳴によって鳴る音とでもいっておきましょうか。順序を追って説明しましょう。倍音列はC，G，C，E，G，B♭，C……と続いています。これは原理もくそも，物理的な共鳴の法則なので興味を持った方は物理の本をご覧ください。でも，音楽だけでい〜やという，僕みたいな楽観的な人は，「ふ〜ん，そうやって共鳴するのか〜。なるほどね〜」と納得しちゃいましょう。え，これ全部覚えるの？ いえいえ，覚えるのはC，G，C（このCは１オクターブ上だよ）。まあ多く見積もってもEまでかな？ さあ，倍音というのは，同時に共鳴しまたは同時

**表3 Cからかぞえた各音程一覧表**

| 音階 | D | D# | E | F | F# | G | G# | A | A# |
|---|---|---|---|---|---|---|---|---|---|
| | | E♭ | | | G♭ | | A♭ | | B♭ |
| 3度 | 減3度 | 短3度 | 長3度 | 増3度 | | | | | |
| 4度 | | | 減4度 | 完全4度 | 増4度 | | | | |
| 5度 | | | | | 減5度 | 完全5度 | 増5度 | | |
| 6度 | | | | | | 減6度 | 短6度 | 長6度 | 増6度 |

ディッシュはこれですね。

それから，借用和音。最初はノンダイアトニックコードといっていたのに，結局，借用和音といってしまうことになりました。どっちも同じことなんですけど，私は今後借用和音ということにします。横文字が好きな人はノンダイアトニックコードとでも読んでください。今回，この借用和音の説明は概要だけになってしまいましたが，基本的なことは理解できたと思います。

さてと，コードの本は巷に溢れてますが，なかなか取っつきにくいことでしょう。だいたい，次から次へと覚えることが出てきて，いったいいつ終わるんだろう，と思ってしまいがちですよね。細かいことを述べてしまうと，覚えることは膨大になってしまって，意気消沈してしまうだろうけど，大まかにまとめると和声で覚えることはあとひとつ。転調についてだけです。簡単じゃないですか!!（大嘘）

＊　＊　＊

先日のZ-MUSICの発売によって，音楽環境がより身近になってきました。私としては，嬉しいことだなと思っていますけど，皆さん使ってみてどうでしょうか？　これが出る頃には，皆さんフィードバックを返してくれているでしょう。え？　おまえはMMLは嫌いじゃなかった？　ま，そうじゃなくて苦手なだけですよ。

でも，フリーウェアにしても，音源ドライバは実に氾濫していてまとまりのない感じだったので，実にタイムリーなものでもあったでしょう。音源ドライバがしっかりしてるから，いまからいろいろ作れるし，SX-WINDOW対応のZ-MUSICを使った楽譜入力ツールなどができそうです。楽しみにしましょう（楽しみにしてくださいでないところがミソですね。根性があれば作ってみましょう）。

どちらにしても，基礎ができたわけですから，応用はいくらでもできるわけです。SX-WINDOW対応の，Z-MUSIC用データセレクタとかね。誰か作って送ってください（他力本願）。

そういえば進藤くんが，SX-WINDOW上のハデな演奏ツールが欲しいといっていました。私も欲しいです（今回願望ばっかいってるな）。それだけ，スタッフはZ-MUSICが気に入っているということですよ。そう受けとめてくださいな。

さて，次回こそはメロディからコードを求めていくテクを話していきたいと思います。そのためには，非和声音についての知識が必要なので，またややこしくなりますね。ごめんなさい。さっきもいったとおり，知識は身につけば足かせではなくなりますから，それらを身につけたうえでの気合の入ったアレンジが送られてくるのを楽しみに待っていますヨ。

それではまた来月会いましょう。

に含まれている音だから，ハモって当然です。**オクターブユニゾン**（１オクターブ上）なら当然だけどＧ（**５度の音だよ**）もハモります。

さて事実。弦楽器のような減衰しにくい音を同時に１オクターブ上で鳴らしても，うまい具合に同じ音色ならかなり耳がいい人でも気がつくことができません。倍音列は先に進むほどハモりにくくなるのですが，Ｇまでならかなり耳がいい人でも気がつきません。ところが，その先のＥになると，同時にほかの音が鳴ってるんだなと，極めてうまく同時に鳴らしても，耳がいい人は気がついてしまいます。慣れてしまえばすぐそのことについてわかるようになるでしょう。

実は，数百年前までＥ，すなわち３度の音は協和音として認められてなかったのですよ。Ｃ，Ｇ，Ｃをよぉ～く見てください（楽譜参照)。１オクターブ上がハモるなら，こんなことをいうことができます。この場合，Ｇは５度の音ですよね。ではＧから数えてその上のＣは何度になりますか？　答えは４度です。逆にいえば，Ｇは下のＣから見ると５度上ですが，１オクターブ上がったＣから見ると，４度下なんですよね。完全音程というのは，このように転回しても，５→４，逆に４→５のように，長音階のスケールをはずれないのです。だから，完全～度というのは１度，４度，５度，８度しかいわないんです。

これが，２度，３度，６度，７度になるとどうなるか？　３度の音は上のＣまで（以後これをひっくり返すといいます）６度より半音１つ分だけ低い音になります。２度もそう，６度も７度もそう。完全以外の音はひっくり返すと長音階のスケールからはずれてしまうんです。長○度といういい方が非完全音程（２，３，６，７度）のうち，長音階のスケールにはまるものと定義すれば，短○度は最初のポイントで見たように，長音程より半音低い音になります。これらを短○度と呼びます。短○度とつくのは２，３，６，７度だけです。

ここでポイント２つ。

１．完全音程はひっくり返しても完全音階になる（長音階からはずれない)。

２．非完全音程（長，短）は，ひっくり返すと，長音階からはずれてしまう。また，長音程なら短音程に，長音程なら短音階になってしまう。

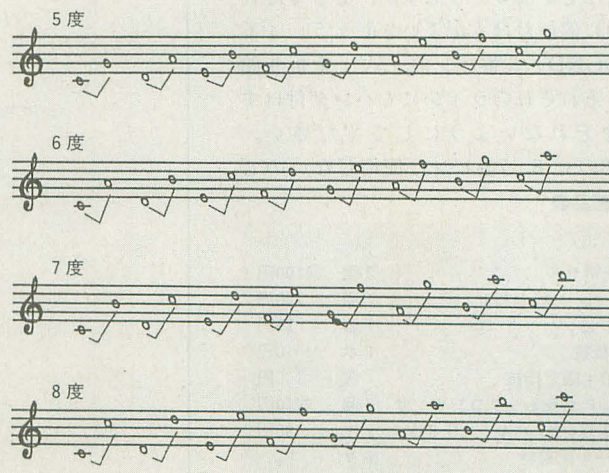
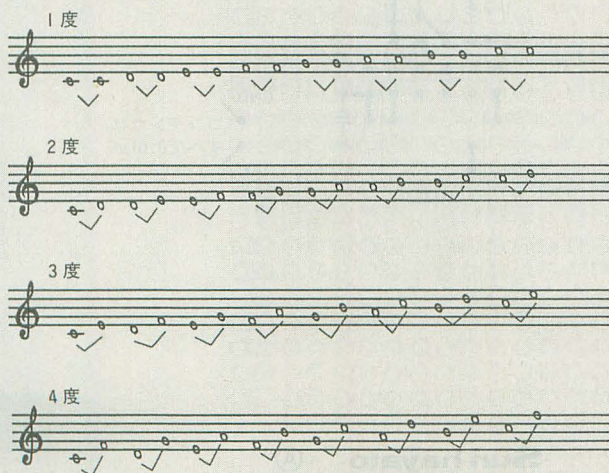
さて，○度といういい方ですがさっき説明した以外にも長短まとめてという意味あいがあることに気をつけましょう。３度の音とは一般に長３度も短３度も指します。

増，減の説明ですが最初のポイントを見てみましょう。増，減は完全音程ならそれよりも半音上下，非完全音程なら短より半音低いのが減，長より半音高いのが増となります。言葉で説明すると余計にややこしくなりますので，次のように表にしてみました。３度から４度まで，このなかにポイントはすべて凝縮されていますからよく見て覚えてくださいね。

スケールキーはＣです。

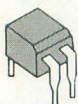
さあ，度数についてのみ込めたでしょうか。連載１回目は度数は暗黙の了解で書いていました。これを読んで理解できたら，即，第１回目にゴー！

**図10 音程の数え方（度数）**

ハードウェア工作入門≪19≫

# ハイテクタンク製作(完成編)

Misawa Kazuhiko
三沢 和彦

5カ月にわたって連載してきた，ウルトラハイテクタンク「パトリオット」の製作もようやく一段落しました。そして，完成したパトリオットが，懐中電灯を持った皆さんのあとを忠犬のように追いかけていることでしょう。

ついに自動追尾ハイテクタンクのパトリオットが完成です。5カ月間，読み続けてくれた読者の方々に感謝します。先月は自動追尾の概念を説明し，光センサーを2個組み合わせた回路で光源の自動追尾を行うシステムを設計してみました。今月はセンサーシステムの工作実習と併せて，自動追尾制御プログラムを紹介します。

## センサー回路の工作

部品表は表1のとおりですが，部品点数は大変少ないので工作も楽だと思います。CdS光電セルについては，T-ZONEパーツショップで入手することができませんでしたので，秋月電子にて購入しました。そのほかの部品は，すべていつものT-ZONEパーツショップで揃います。

では順番に基盤に取り付けていきましょう。図1の実体配線図を参考にして作業を進めてください。なお，実体配線図はハンダ付けする側から見た図であり，部品を取り付ける側からではないことを改めて確認しておきます。

初めにICソケットを取り付けます。今回は基盤に余裕があるので，どこに取り付けてもなんとかなりそうですが，できるだけ真ん中に置いたほうがよいでしょう。4番ピンのGNDと8番ピンの+5Vとを折り曲げて，それぞれのラインにもハンダ付けするのを忘れないようにしてください。LM393の5，6，7番ピンは使用しないので，

表1 部品表

| 品名 | 数量 | 価格 |
|---|---|---|
| CdS光電セル | 2個 | @100円 |
| コンパレータLM393 | 1個 | 100円 |
| 8ピンICソケット | 1個 | 25円 |
| 1kΩ抵抗 | 1本 | 10円 |
| 10kΩ半固定抵抗 | 1個 | 100円 |
| 0.01μFセラミックコンデンサ | 3個 | @10円 |
| IC用基盤(サンハヤトICB-87) | 1枚 | 90円 |
| ビニール配線材 | 少々 | |

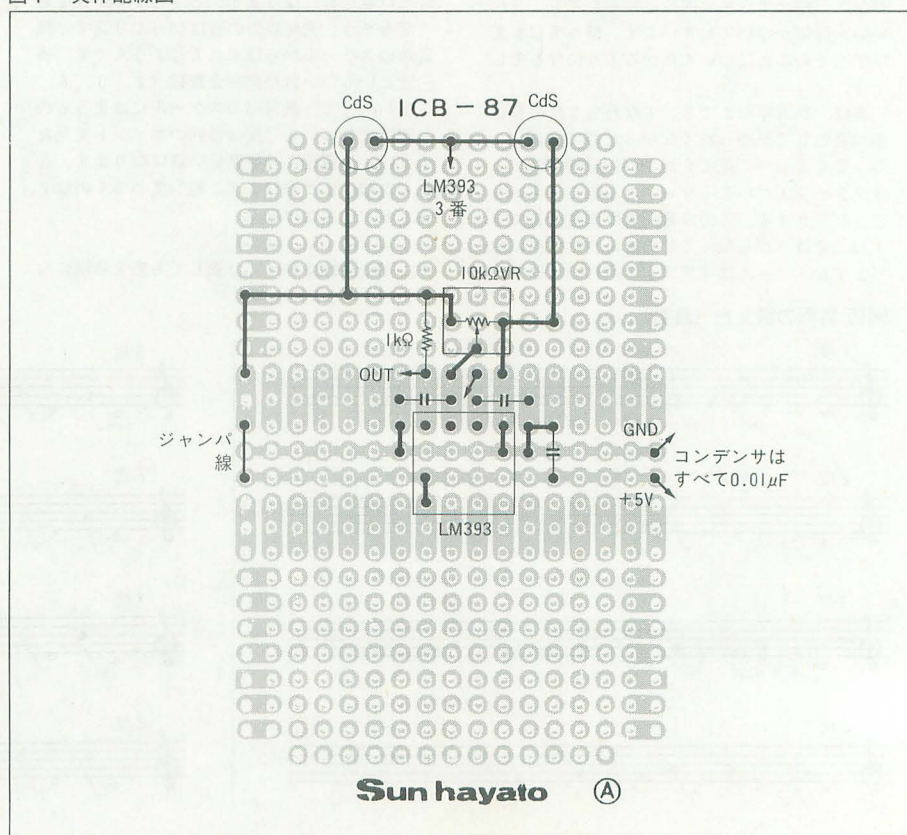
そのままに放置しておきます。

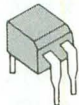
次に半固定抵抗の位置を決めます。半固定抵抗は実体配線図中にあるとおり，3本の端子が出ていて，そのうちの2本が固定抵抗，残り1本が中点になっています。中点端子はLM393のマイナス入力(2番端子)につながるのですが，バイパスコンデンサの配置の都合上，3番ピンのところから斜めに接続しています。半固定抵抗の端子を斜めに折り曲げて，2番ピンのところにハンダ付けしてしまいます。もし長さが足りなければ，あとでバイパスコンデンサを取り付けるときにその足を折り曲げてハンダ付けします。

そして，両端の固定抵抗の端子はとりあえずそのままにしておき，次は1kΩ抵抗を取り付けます。その足を折り曲げて，片方を先ほどの半固定抵抗の足にハンダ付けします。

CdS光電セルは，そのまま「パトリオット」の目として機能するので，ほかの部品から少し離して配置します。さらに，2つの光電セルの間も間隔をおく必要があります。今回，秋月電子で入手したCdS光電セルの足がかなり長かったので(2.5cm)，そのまま切らずに折り曲げてハンダ付けしました。右目のほうは，半固定抵抗の両端の端子の片方を経由してGNDに落とします。左目のほうは，1kΩ抵抗と半固定抵抗のもう片方の端子との両方を経由して+5Vにつなぎます。

注意してほしいのは，この配線を見ると

図1 実体配線図

先月説明したコンパレータの動作と逆になっている点です。すなわち，パトリオット本体が光源に対して左に向きすぎていると，プラス端子の入力電圧は低いほうにシフトし，その結果，出力はLということになります。逆に右に向きすぎるとプラス端子の電圧は高いほうにシフトし，出力はLになるというわけです。しかし，実際問題としては，制御プログラムの中でソフトウェア的に対処できますから，あとで述べる動作チェックのところで確認すればなんの問題もありません。

最後にバイパスコンデンサ3個とジャンパ線とを取り付け，最後にDIN中継ジャックから出ている＋5V，信号線，GNDのビニール線を基盤にハンダ付けすれば完成です。ジャンパ線は2個所あり，2個のCdS光電セルの中点とLM393の3番ピンを結ぶ線および，左目のCdS光電セルから＋5Vにつなぐ線になっています。

## 回路のチェックと調整

まず，リスト1を使って光センサー回路のチェックをしてみましょう。今回工作した回路は，すでにモーターからのリード線と合わせて，中継ジャックにつながっていると思います。インタフェイス基盤のDINプラグとつなぎ，そのインタフェイス基盤をいつものように汎用フラットケーブルを使いX68000本体のジョイスティックポートに接続します。

ちなみにリスト1は単純にジョイスティックポートの入力を読んでくるだけです。このチェックプログラムを実行させる前に，小さなマイナスドライバで，半固定抵抗の位置をおよそ真ん中に合わせておきましょう。半固定抵抗は上から溝の部分にマイナスドライバを差し込み，回すことによって中点端子の抵抗値を変えることができます。回せる範囲はいちばん安い簡単なタイプで225°程度ですが，ある程度高級な多回転型（以前のフォトダイオードによる光量計で使ったもの）であれば，6回転近く回してやっとフルスケールとなります。今回はいちばん安いタイプを使っています。

リスト1

```
 10 /* save "d:¥basic¥eyecheck.bas
 20 /*
 30 /*光センサー動作チェック用プログラム
 40 /*  1991.11.16  K. Misawa
 50 /*
 60 int v
 70 while 1
 80   /*最下位ビットを読む
 90   v=(ioinp() and 1)
100   print v
110 endwhile
120 end
```

このように半固定抵抗をラフに合わせたあとにリスト1を実行させてみましょう。両方の光電セルを部屋の明かりのほうに向け，センサーを片方ずつ指でさえぎってみます。このとき，右を隠した（左の光量が多い）ときに1，左を隠した（右の光量が多い）ときに0となっていれば動作チェック完了です。

これだけ簡単な回路なので，ほとんど一発で正常動作することと思いますが，万一うまく動かないときは，配線ミス以外ありません。IC周りのハンダ付けが少し込み合っているので隣とくっついていないか，ジャンパ線は忘れていないかなど，実体配線図と照らし合わせて十分にチェックしてください。

また，延長ケーブル間の配線ミスも注意してください。ジョイスティックポートとインタフェイス，そして今回の回路をすべて接続したうえで，LM393に＋5Vがきちんとかかっていなければ話になりません。さらに，LM393の1番ピンの出力信号がジョイスティックポートのIOA0端子に伝わっているかもチェックしてください。

以上，怪しいと思われる点を列挙してみましたが，皆さんはなんの問題もなく動作させられたことと思います。

さて，正常に動作することを確認したら今度は調整です。調整は先ほど述べた半固定抵抗の位置を動かします。できたら懐中電灯を用意してください。部屋の明かりを暗くし，2個のCdS光電セルに均等に光が当たるように懐中電灯を向けます。ここで，リスト1を実行させたあと，半固定抵抗を動かしてみます。半固定抵抗のある位置を境に表示の0と1が入れ替わりましたか？すでに片方ずつの光電セルを交互に隠したときに0と1が入れ替わるのを確認してあれば，必ず，半固定抵抗を動かしたときにも0と1が入れ替わるところが見つかるはずです。

調整は，ちょうど0と1が入れ替わる境界のところに半固定抵抗をセットするのです。適当なところでセットできたら，懐中電灯の向きを左右に振って，光電セルの光の入り方のバランスを変えてみましょう。左右に振って，やはり0と1が入れ替われば調整完了です。もし思ったように入れ替わらないときには，図2に示すように2個の光電セルの間についたてを入れてみるといいでしょう。私は黒い紙を張り付けました。こうすると，光を傾けたときに反対側の光電セルはついたての陰になり，光の左右のアンバランスが強調されます。

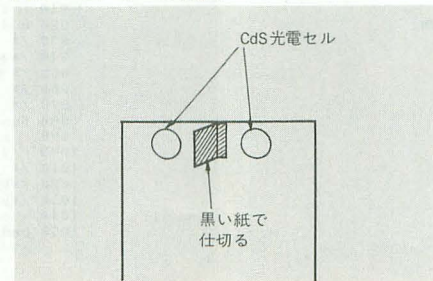
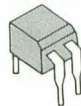



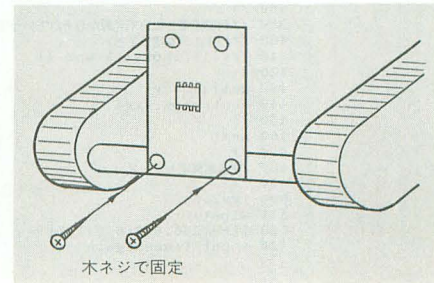
図2　センサーの配置

## 自動追尾制御プログラム

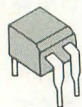
いよいよパトリオットに自動追尾をさせてみます。工作した基盤を木ねじ2本でパトリオット本体の前面に取り付けます（図3）。部屋を暗くして，部屋の隅に懐中電灯を置き，光をパトリオット側に向けます。リスト2を実行させると，"Stand By OK?"と尋ねてきますので，モーター用の電源をONにして，パトリオットを床に置いて発進待機状態にしておきます。

そして，キーボードの任意のキーを押すと首を振りながら光の位置を探して進んでいきます。意地悪く，途中で光源の位置をゆっくり動かしてみましょう。それでもパトリオットは自分の向きを変え，光源を追っていきます。また，光源を見失ってもできるだけ明るいほうを探して動きますので，もしかすると今度はパソコンのCRTに向かっていくかもしれません。

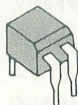
もし最初から光源を追尾せず，むしろ光源から逃げるように動いてしまうときは，設定が逆なので，リスト2中の指示に従って命令文を入れ替えてください。

では，制御プログラムの中身をもう少し詳しく追ってみましょう。基本的には，11月号のプログラムに今月号のリスト1を加えたようなものです。変数設定，画面設定後，メインルーチンに入ってから，FOR～NEXTループで光センサーからの読み込みと，モーターへの出力を交互に繰り返していきます。光センサーの入力0/1を制御コマンドに変換する式が370行にあります。

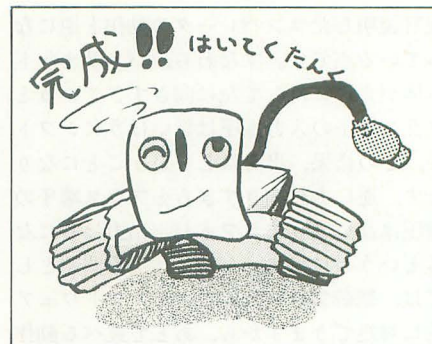
図3　基盤の取り付け方

動作は，右に向きすぎていると判断したら左旋回，左に向きすぎていると判断したら右旋回させるだけの簡単なものです。常にどちらかに旋回していて直進がないので，動きがぎこちなく見えますが，自動追尾には問題ありません。

さて，制御コマンドを出力したあとにdisplay関数でグラフィック画面に出力しています。グラフィック画面は1ステップごとにパトリオットがどう進んでいるかをトレースしていきます。そこで，出力した制御コマンドを順番に配列変数にでも記憶させておけば，パトリオットの動作を再現することもできます。今回のプログラムではトレースデータを配列dat( )に格納しておき，1回走行したあと，“Trace Again ?”の表示を出して同じ動作を再現するようにしています。このように，デジタルデータとして取り込んでおくことによって，動作を再現できる点がコンピュータ制御の強みともいえましょう。

## まとめ

以上5カ月にわたって解説してきたとおり，自動追尾システム搭載，前後進左右旋回可能なスーパーハイテクタンク「パトリオット」が完成しました。説明の中でかなり重複する部分もあったかとも思いますが，やる気さえあれば，Oh！Xの読者の誰もが「パトリオット」を製作できるように配慮したつもりです。

そして，自動追尾システムとはいっても，この連載でこれまでに説明してきたことの積み重ねでしかないことがわかってもらえたと思います。ハードウェア工作がより皆さんの身近なものになれば幸いです。

## 次回作は？

さて，入門記事としては超大作を完成させたあとは，どうしようかと迷っていますが，読者の方からそろそろジョイスティックポートにくっつける形の作品ではなく，単体で動作させられるものに移行していったらどうかという意見をいただきました。

現在のところ，コンピュータの基礎から学べるように，独立したマイクロコンピュータボードを設計製作していくか，あるいはホビー向けの愉快な実用工作をいくつか紹介していくか，などを計画しています。より多くの読者の意見を取り入れて方向を定めていきたいと思いますので，読者ハガキの隅にでも意見を書いてもらえるとありがたく思います。

ところで次回はまだまだジョイスティックポートにつなぐインタフェイス回路が登場します。今回リモコン操作を実験してみましたが，ケーブルでつながっているのはカッコ悪いという意見がありました。そこで……？ こう書くと，皆さん想像つきますね。赤外線リモコンを使い，データを無線で飛ばす回路を設計製作してみようと思います。あくまでも入門編ということなので，できるだけ簡単にしかも応用性のある回路にしていこうと思います。楽しみにしていてください。

**リスト2**

```
  10 /* save "d:¥basic¥patriot.bas
  20 /* save@"d:¥basic¥patriot.doc
  30 /*
  40 /*パトリオット自動追尾制御プログラム
  50 /*
  60 /*     1991.11.16  K. Misawa
  70 /*
  80 int v=4,nmax=1000,n,nstep
  90 int xx0,xx,yy=256
 100 dim int dat(512)
 110 int flag=0
 120 /*光源までの距離に応じてstepの値を変える
 130 int dist,step=3
 140 str z
 150 dist=int(512/step)
 160 n=nmax*0.9#
 170 /*
 180 /*画面設定
 190 screen 1,1,1,1
 200 console,,0
 210 window(0,0,511,511)
 220 apage(0)
 230 vpage(15)
 240 /*
 250 /*メインルーチン
 260 /*
 270 while 1
 280 outval(4)
 290 yy=256
 300 wipe()
 310 line(256,0,256,511,7)
 320 input"Stand By OK";z
 330 for xx0=0 to dist
 340 /*
 350 /*光センサーからの読み込み
 360 /*
 370 v=2-(inpval() and 1)
 380 /*
 390 /*自動追尾しないで光源からそれていってしまう時は
 400 /*以下の式を使うこと
 410 /*v=1+(inpval() and 1)
 420 /*
 430 dat(xx0)=v
 440 control(v,xx0)
 450 /*
 460 next
 470 /*
 480 /*追尾再現トレース
 490 /*
 500 yy=256
 510 wipe()
 520 line(256,0,256,511,7)
 530 input"Trace again";z
 540 for xx0=0 to dist
 550   control(dat(xx0),xx0)
 560 next
 570 /*
 580 endwhile
 590 end
 600 /*
 610 func control(v,xx0)
 620 outval(v)
 630 display(v,xx0)
 640 /*
 650 for jjj=0 to n: next
 660 /*
 670 outval(4)
 680 for jjj=0 to nmax-n: next
 690 endfunc
 700 /*
 710 /*データ入力
 720 /*(引数)なし
 730 /*(戻り値)ジョイスティックポートの値/*
 740 func int inpval()
 750   int v
 760   v=&B1111-(ioinp() and &B1111)
 770   return(v)
 780 endfunc
 790 /*
 800 /*データ出力
 810 /*(引数)整数値
 820 /*(戻り値)なし
 830 /*(機能)引数を8で割った余りを出力
 840 /*
 850 func outval(d0;int)
 860   int v,v0,v1,v2
 870   v0=1-(d0 and 1)
 880   v1=1-(d0 and &B10)/&B10
 890   v2=(d0 and &B100)/&B100
 900   v=&B10000000*v1+&B1000000*v0+&B10000*v2
 910   ioout(v)
 920 endfunc
 930 /*
 940 /*トレースディスプレイ
 950 /*(引数)制御コマンド v
 960 /*       ループ変数 xx0
 970 /*
 980 func display(v,xx0)
 990   xx=511-xx0*step
1000   yy=yy+(2*v-3)*8
1010 /*
1020 /*行の式を入れ替えた時はこちらの式を使う
1030 /*yy=yy-(2*v-3)*8
1040   circle(yy,xx,2,15)
1050 endfunc
```

# 明日に向って3.5インチ

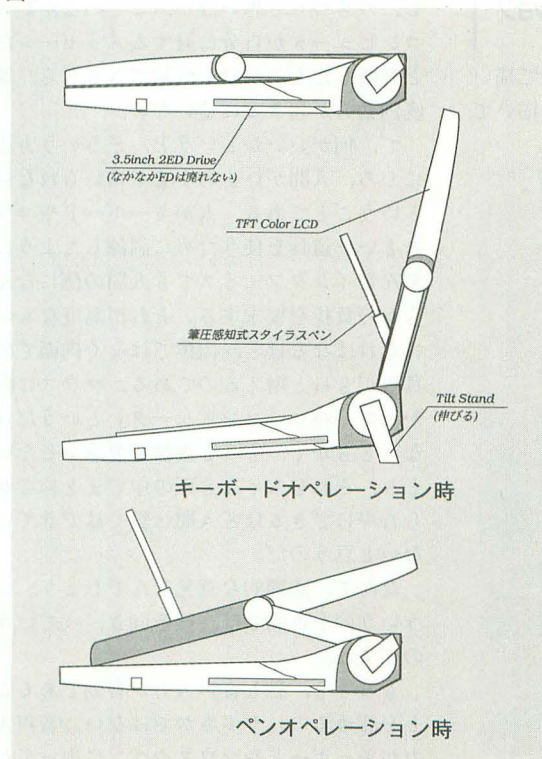
Ogikubo Kei 荻窪 圭

今月は理想のパソコンの姿を探る，という長い前置きから始まり，3.5インチディスクの話へと進んでいきます。パソコンの未来とかって新鮮味のないネタかもしれませんが，初夢ということで楽しんでください。

いやあ，先月号の「Oh!X INDEX '91」を見たのだが，「大人のためのX68000」のタイトルってひどいね。統一性もないし。来月からはちゃんと考えよう。

さて，コンピュータというものについて考えるときに，アメリカでも日本でも“いままでコンピュータを使っていなかった人にいかに使わせるか”がひとつのテーマになっている。で，そのひとつがペンベースコンピューティングってやつ。

## ペン入力について考える

ペン入力の何がいいかっていうと，入出力面が一致していることだ。マウスはペン入力にいたるまでの過渡期だという人もいるくらい。で，ペン入力するなら，スクリーンが目の前に立っていては腕が疲れるわけで，当然寝かせて使うことになる。が，文字をどんどん入力するだけならキーボードのほうがいいわけで，できれば両方ほしいところ。

というわけで，荻窪圭は考えた。

これは，ノートブックコンピュータのひとつの姿である(図1)。合体変形ロボットみたいになってしまったのは愛嬌だ。キーボードベースで使うときは，ペンはマウスのようにポインティングデバイスとして働く。そのくらいなら，画面が多少立っていてもかまわない。

で，ペンベースで使うとき（図形を扱うときや，文書の編集など）は3番目のように，ディスプレイ面がずりっと手前にスライドするわけである。

どうせ未来（とりあえず3年後）の話だから，CPUは68040の25MHzで，メモリは標準で8Mバイトで，ディスプレイはTFTのカラー液晶で，ハードディスクは内蔵で170Mバイトだな。重さは2kg。バッテリは水素ニッケルでもなんでもいいけど，5時間くらいは使いたい。

図1

そうそう，スタイラスペンの筆圧感知式というのはもうできていて，マウス以上の表現も可能だ。強く書くと太い線になったりとかね。

可搬メディアには一応3.5インチFDDをつけてはみたが，本当のところは，フラッシュメモリといきたい。でも，きっと高いはずだ。もちろん，イジェクトボタンなんかはなし。

で，こうなるとOSというか，シェルもそれなりのものが必要になるし，Pen-PointやWindows for Penのようなジェスチャという編集動作のシステムも必要になるだろう。

もちろん，携帯電話は内蔵していて，通話も通信も可能だ。

これで，30万円！

さて，問題は，これで何をするか，である。私は，ビジネスマンのためのコンピュータを考えているわけではないから。

## デスクトップマシンは?

翻って，デスクトップマシンについても触れざるをえない。どーせだから，と，最近流行のディスプレイ一体型にしてみた。デザインはフィリップスが出したヘルメット型のテレビの真似。

いや，なに。これが大傑作だといっているわけではない。このくらいバカなデザインにでもしないと，駄目なのである。特に，最近のディスプレイ一体型マシンたちを見るにつけ，そう思う。

簡単に説明すると，ディスプレイはCRTというのもなんなので，TFTカラー液晶にしてみた。ハイビジョンというのも考えてはみたが，でかくなりそうなのでやめた。液晶にしては奥行きがあるが，そこは気にしないように。廉価版でCRTバージョンもあることにしよう。

外部記憶装置はCD-ROMと3.5インチFDD（2EDかもしれないし，フロプティカルかもしれないけど）をひとつ。3.5インチ光磁気かもしれない。なんでもいいや。ただ，3.5インチFDはなかなか廃れないと思う。つけ忘れたけど，ICカードスロットも必要かもしれない。ハードディスクは内蔵で，300Mバイトくらいあればいいのではないだろうか。

で，バイザーは，一応目にいいのだ，といっておこう。反射が気になるときは，降ろすわけね。

本体下にあるのはチルトスタンド。左右360度，上下は15度くらいかな。チルトする。黒いのは，チルトの固定ボタン。この360度回転というアイデアはPS/55Zからいただいた。こいつは便利なのに，PC-9801CSもFM TOWNSII UXも採用しなかった。ちなみにアゴのあたりにあるのは，本体を回したりするときに便利な把手である。

せっかくのヘルメット型だから，マイクもつけておいた。もちろん，ボイスナビゲーションはサポートしている。

前面にスイッチやコネクタ類はいっさいない。ビデオ入出力や音声入出力コネクタくらいはカバーの中につけておいたほうがいいかも。

電源スイッチは，というと，キーボードにある。ジョイスティックは，というと，MacintoshのADBバスの知恵を拝借した。つまり，キーボードもマウスもジョイスティックもみんな同じコネクタで同じバスなのだ。だから，専用のコネクタはいらない。キーボードの後ろからでもとればいい。こいつのおかげで，マウスの代わりにペンはつながるし，複数のマウスをつないでも別のものとして認識してくれるのだ。

CPUはなんでもいい。速いRISCでも68040でもいい。

となるとOSであるが，5年後ということで，IBMとアップルが共同開発するとかいうPinkでもいいや。NeXT STEPだったりして。Windows NTはちょっとやだな。

その他，つらつら思いつくままに挙げていくと，テレビはもちろん標準で，ハイビジョン放送も見ることはできる。もちろん，画像取り込みや画像書き出し（？）は自在だ。丸いと持ちづらいだろうが，頭頂にはもちろんポップアップするハンドルがある（CRTを使っていないからそれほど重くない)。頭の両側に，機銃をつけてもいいかな（何に使うんだ！）。

さらに，空いたスペースにはレーザープリンタのエンジンが入る。レーザープリンタをつけるときは，排紙のために本体とチルトスタンドの間に1cmくらいのペーパーユニットが入る。

周辺機器を増設するときも，ケーブルはいらない。128チャンネルの無線だ。自動的に空いているチャンネルでつながる（ついでに，その電磁波で奇形児も急増だ！）。

内部にデジタル通信のインタフェイスを持っており，デジタルのセルラー回線を使って遠隔通信も可能だ。

ついでにレジューム機能と，ACからの電源供給がなくなっても1時間くらいはRAMの内容を保持できるバッテリーくらいはつけておきたい。

忘れてはならないキーボード。たとえば，ファンクションキーはいらない。INSキーもいらない。とにかく，コンパクトなものがいい。矢印キーの配列については，X1みたいにならないよう，考えることにして，保留。とにかく，キーボードはコンパクトがいちばんである。

マウスについては悩んだのだが，思いつかなかったので，しゃあない。

この上位機種として，球形のディスプレイユニットの下にピザボックスタイプの本体という組み合わせもある。このタイプだと，ディスプレイが90度回転し，自動的に縦長にも横長にもなるのだ。文章を書くときは縦長に，ふだん，あるいはフルスクリーンテレビにするときは横長にするのがいいだろう。

## 新しいコンピュータとのコミュニケーション

ペン入力。手書き文字認識。自分で描いたメモでさえ，2，3日すれば何が描いてあるか判読できなくなるのに，コンピュータがそれを読み取ってくれるわけが……そうなったりして。何日もかけて学習させて，もしかして，自分でさえわからない殴り書きをコンピュータが微妙な癖を読み取って，ちゃんとテキストにしてくれたりして。書いた当人でさえわからないものを読み取るなんて，恐ろしいなあ。

それにはまず，筆圧まで細かく検知するペンと，ペンのすべての動きを記録しておくだけのメモリが必要だろうな。

音声？ コンピュータに向かってしゃべる？

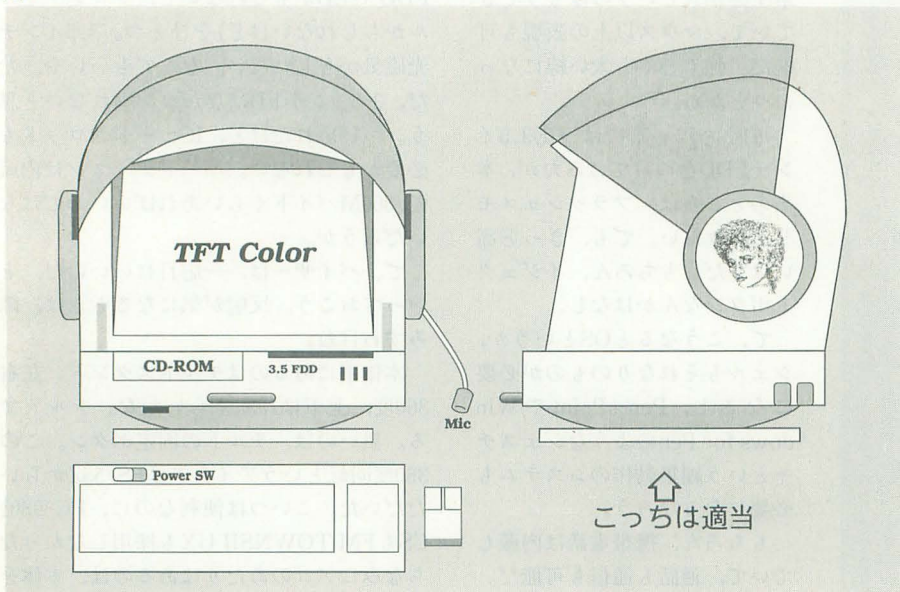
最近の人類は機械に向かって，というより，人間以外のものに向かって話しかけるのに慣れているようだから，問題はないかもしれないが，私はいやである。キーボードやマウスだから，こちらのメッセージがコンピュータに届くように気を使ってあげられるのであって，そこまで気を使わなければならないなんてのはいやである。

よく，頭で考えたことを読み取ってくれるコンピュータがあればいいという話を聞くが，よくもまあそんなに恐ろしいことを，と私は思うわけだ。だって，考える，っていうのはどういうことだ？ 的確に，コンピュータへのメッセージだけを考えるということ（頭に思い浮かべるということ）が人間に可能だろうか。そこまで自分を訓練したいとは思わないし，そんなことをしたら，ストレスが溜まってしようがない。もし，つらつらと思い浮かぶことのなかからコンピュータが自分に対するメッセージだと判断したものだけを解釈できるなら，家庭内暴力が起きるに違いない。

で，何がいいかというと，どういう方法にしろ，人間がいまのままではいられないということである。人がキーボードやマウスという道具を使うために訓練したように，どんなインタフェイスでも人間の側になんらかの負担を要求する。それが高度なものになればなるほど，肉体ではなく内面での負担がもっと増えるのである。マウスに向かって「ハローコンピュータ」というだけならともかく，もっと複雑な要求をしたいとき，それをちゃんと頭の中でまとめてから音声にできるほど人間は賢くはできていないと思うのだ。

なんて，悲観的な意見なんでしょう。こういうのを世間では，後ろ向き，っていうのですね。

もっとも，私も音声入力が有効であることを認めるのにやぶさかではない。音声入力がキーボードやマウスやペンに取ってか

図2

わる事態は想像しづらいが，普通の人がコンピュータを使うときはほとんどがルーチンワークだろう。となると，音声入力は非常に有効だ。知的な指示を声で与えようと思うからいけない。どのくらい実用になるかはわからないが，ボイスナビゲータっていう商品もある。

もし，新しいインタフェイスが開発されるなら，バーチャルリアリティなものから音声もの，脳波ものまでひっくるめてゲームにしたいね。ゲーム，アミューズメント，ひとりスターツアーズ。やっぱりそういうものだろう。

プレイヤーの感情を微妙に読み取ったりして，コンピュータ"きもだめし"，なんていうバカなものから，脳波監視つきのダンジョン・マスターまで（こっちの心理状態でキャラクターの動きが変わる）。プレイヤーがよこしまなことを考えると，いきなりパーティ全員の属性がイビルになってしまうウィザードリィとか。んなもん，誰もやんないって。

*　　*　　*

さて，初夢話が長くなってしまったが，そろそろやめて，本題に入ろう。今回の本題は3.5インチディスクのフォーマットである。X68000用の3.5インチFDDも登場したことだし，ということで，ちょっとまとめてみた。表を見てほしい。

率直にいってひどい話である。これだけのいろんなフォーマットがあるのだ。

表の見方であるが，最初の容量はトラック数やセクタ数などの値から計算したフォーマット後の容量。ただし，Macintoshの場合は，ユーティリティソフトにてチェックした値を載せてある。その右の容量は，俗に何と呼ばれているか。トラック数やセクタ数，バイト数はいいよね。容量の計算は，トラック数×2×セクタ数×バイト数だ。つまり，1セクタに512バイト入っていて，1トラックに8セクタ確保してあって，ディスク片面に80トラックあって，両面を使っているから2倍する，と，655,360になり，それを1,024で割るとKバイト単位になるのである。

Macintoshについては，トラックによってセクタ数だかセクタ長だかが変わるらしいので，よくわからない。

採用機種は，その容量をカタログに記載しているもの。使用可能機種は，実は使えるのだ，というものとユーティリティを使えば可能，というものがある。機種に関していえば，IBM PC系のなかにはPS/55なども含まれる（動作チェックはPS/55Z SXで行った）。

それぞれ見ていこう。

まず，2DD。640Kバイトと720Kバイトの違いは，セクタ数だけである。どっちも読み書き可能なら，多いほうがいい。どうしてわざわざ8セクタにしたのだか。ちなみに，ワープロ専用機にはMS-DOSフォーマットをサポートしているものが多いが，多くは720Kバイトフォーマットである。Macintoshを含めて，こいつがいちばん多くのマシンで使えるのだ，ということは覚えておこう。

ただし，PC-9801はタコのため，外付けドライブの2DD/2HD自動認識ができない。だから，PC-9801の外付けドライブの場合は，認識するためのボード付きのドライブユニットでない限り，面倒臭い。

Macintoshの場合，システムに添付される"Apple File Exchange"というプログラムで変換する。

Macintoshの800Kフォーマットについては，まあ，独自の世界である。

続いて，2HDだ。まず，1.2Mバイトの世界から。J-3100もPC-9801も1.2Mバイトだが，実は微妙に違う。PC-9801の場合，NECはいまだに1Mバイトと称しているが，そんなことというやつはいない。1.2Mバイトというのが一般的。

で，両者の違いはセクタ長にある。FDDはセクタ単位でデータを読み出すわけだから，セクタ長が長いほうが速い。およそ，2倍である。だから，PC-9801の採用したフォーマットのほうが，速いわけである。

どうしてNECが片面80トラックにせず，77トラックにしたのだろうか。5インチとの互換性のためだろう。

IBM PCというか，PS/2の採用した2HDは1.44Mバイトであり（単純に，720Kバイトの2DDを倍にした），容量が多い。こいつは1.2Mバイト用のFDDでは読めない。最近はDynaBook VやPS/55などで，両方使えるドライブを搭載したものが増えてきて，ラッキーである。Macintoshでは，2DDと同じく"Apple File Exchange"というプログラムで変換できる。

2EDというのは東芝が開発したフォーマット。NeXTで採用された。IBM PC版のDOS 5.0でも使えるようになっているが，サポートした国産ハードウェアの話はまだ聞かない。未知数である。

X68000に外付け3.5インチドライブをつけるなら，ぜひとも，720Kバイトのディスクを読み書きできるものにすべきである。ということだな。

ツクモ電機オリジナルの3.5インチ2HDドライブはどうやら，1.44Mバイトも1.2Mバイトもいけるやつらしい。もちろん2DDも可能で，日本IBM風にいえば3モードドライブというやつだ（ドライバで対応している）。MacintoshやAMIGA以外のフォーマットならOkというものだね。なんにしろ，ツクモオリジナルのわりには，ツクモ特価があったりして不思議である。ツクモのオリジナルといえば，限定99台のMacintosh用リムーバブルハードディスクというやつがあった。ちなみに，編集のA氏がAMIGA用に買ったのがシリアルナンバー22で，私が買ったのがシリアルナンバー24だった。ちゃんと99台売れたのだろうか。

で，3.5インチだが，X68000からほかのマシンにデータを持っていったり，ほかのマシンからファイルを持ってきたりするときには，ファイル名にくれぐれも気をつけるように。最近やりがちなのが，DOSマシンやMacintoshからデータを持ってくるとき，思わずファイル名にハイフンをつけるというやつだな。さすがに，"小文字ファイル名"というのは私でさえやらなくなった。

以上。長い前置きのわりには短い本題であった。今回は何の話をしたかったのだっけ。全部，Press Conductorが悪い。来月こそは行くぞ。

表1

| 使用ディスク | 容量(KB) | 容量（俗称） | トラック数（片面当たり） | セクタ数（トラック当たり） | バイト数（セクタ当たり） | 採用機種 | 使用可能機種 |
|---|---|---|---|---|---|---|---|
| 2DD | 640 | 640KB | 80 | 8 | 512 | PC-9801他 | J-3100，IBM PC系 |
| 2DD | 720 | 720KB | 80 | 9 | 512 | IBM PC系 | PC-9801，Macintosh |
| 2DD | 797 | 800KB | — | — | — | Macintosh | |
| 2HD | 1200 | 1.2MB | 80 | 15 | 512 | J-3100 | PC-9801，PS/55の一部 |
| 2HD | 1232 | 1.2MB | 77 | 8 | 1024 | PC-9801 | J-3100，PS/55の一部 |
| 2HD | 1440 | 1.44MB | 80 | 18 | 512 | IBM PC系 | J-3100の一部，Macintosh |
| 2HD | 1437 | 1.4MB | — | — | — | Macintosh | |
| 2ED | | 2.88MB | | | | NeXT | |

# 山越え，谷越え，どこまでも（前編）

プロジェクトチーム DōGA　かまた　ゆたか

**連載の最終回を締めくくるため，当チームの作品制作の実況レポートをお届けします。諸処の都合により，前後編となってしまいましたが，前編はおもに作品企画編です。**

## はじめに

この号が出る頃には，芸術祭の地区予選はすべて終了していると思うが，大阪の地区予選が11月10日に行われた。私たちも一応グラフィック部門で，つまらないCGAをひとつ出品させていただいた。そのへんの話は，あとで詳しく報告するが，実は，初めはゲーム部門でエントリーしようとしていたのだ。

どういうゲームかといえば，市販のあるゲームを元にしたもので，起動直後の画面は，赤いドロドロしたものが画面いっぱいに描かれる。しばらくの間は，どのような操作をしても，なにも状況は変わらない。しばらくの間といっても，数秒とか数分なんてなまやさしいものではなく，数時間，数日，数年……いやいや数百万年間この画面が続く……。そう，このゲームは，リアルタイム版シムアース！（それじゃ一発ギャグ部門だって）

＊

さて，最終回は，作品制作の実況レポートをお届けしようと思ってましたが，ハッハッハッ，情けないことに，ついに新作は完成するに至りませんでした（お倉入り）。そこで，前回の予告など（例によって）忘れたフリをして，全然別の話でもしようかなとかも考えたのですが，完成しないというのも，CGA作品制作ではよくあることですので，そのあたりの原因など追及するためにも，挫折するに至った道のりを克明に記そうと思います。皆さんの作品制作の参考にしてください。まぁ，CGA作品制作なんて，反省，立志，挑戦，挫折（ときどき成功）の繰り返しですよ。

## まずは過去の反省

ずいぶん前（1990年7月号）になりますが，作品企画について，当チームの実例などを紹介しました。その後，あのときの“グランドキャニオンで複葉機がF16とドックファイトする話（仮名：がんばれスキャット！）”はどうなったのかという問い合わせが何件かありました。残念ながら，試験的に数カット作っただけで，完全にお倉入りになっています。原因はいろいろ考えられるのですが，監督が上級生の支援をあてにしすぎた，逆にいえば上級生は口を出すだけで手伝ってやらなかったというあたりに問題があったようです。そこで教訓，

## 柚姫の明るい悩み相談室

お久しぶり，柚姫です。皆さんお元気ですか。姫は，なんかこのところさぼり癖がついちゃって……。久しぶりにお仕事だ～，と思ったら，なんか連載が終わるとか，終わらないとかいってるし，う～む。

前回で「まだ少し余っているので，どうしてもいますぐCGAシステムが欲しい人は「面倒くさいCGAシステム配布係」まで申し込んでください」と書いたところ，びっくりするほどたくさん申し込みがありました。もうすぐ次のバージョンが出るのにね。

では，システム申し込みのお手紙の中からいくつか紹介します。

＊

Q：CGAシステム希望！　親切な宛名ラベル（3種）を同封します。
A：気を使っていただきまして，どうもありがとうございました。でもねー，残念ながらシステムは，“ゆうパック（カーボン複写）”で送るんですよ。
Q：カンパが少なくてすまないと思う気持ちでジャズインでも買って飲んでください。うまいから（変な文）。
A：よくわからないよー。ジャズインってなんですか。今度見つけたら飲んでみます。ところでこの頃，姫は味覚がおかしいといわれます。今年キャンプに行ったときのこと，うどん玉4つ，チキンラーメン4つ，なのに鍋がひとつしかなくて，姫はしかたなく（迷わず？）うどんとチキンラーメンを混ぜて作りました。みんなおなかいっぱい食べたくせに文句いうんですよー，プンプン!!　けっこうおいしかったけどなー。
Q：ただの手紙なので，食べてもおいしくありません。バグレポートでもないので，怖くありません。
A：食べれる手紙も食べれない手紙も，怖い手紙も怖くない手紙も，お待ちしています。でも食べれる手紙がいちばんいいな，味はお好み焼き味で（おなかのすいた姫）。
＝この間，お好み焼きを食すため40分中断＝
Q：下宿でインコを飼おうと思います。どんなインコがいいのでせう？　どなたか詳しい人がいたら教えてください。
A：DōGAはペット屋じゃないんだけどなー。なんでこんな手紙がくるのかな～。でも姫は個人的におかめインコなんか，あの丸さ加減が好きです。でも高い。やっぱり安くて丈夫なのは普通のインコかな。姫の家に昔いたインコはとまり木にとまれなくて，床の上を歩き回っていました（ときどき背中にほかのインコのフンが乗っていた）。
Q：家を購入したばかりなのであまりカンパできませんが，よろしくお願いします。
A：いえいえ，こんなにたくさんいただいてしまってすみません。これが本来なら柱の1本，窓ガラスの1枚になっていたのかと思うと……。家が崩れたりしないことをお祈りしてます。
Q：芸術祭で「あ！」と「あ！ Ver.2」を作った蟻馬です。11月10日の学祭にCGAをやりたいと思っていますので，早く送ってください。
A：あ！　もう間に合わない。ごめんなさい。

＊

前にカンパでサニーレタスを送っていただいた方から，今度はたくさんの林檎を送っていただきました。今年特に林檎が不作で高い!!　本当にありがとうございました。おかげさまでモデラー高津さんは，2日間林檎だけを食べて生きてました。

**1　自分が作りたい作品を作れ**

人はとやかく口を出し，批評する。しかし，口を出すやつほど，実際には手伝ってくれないものである。そんな無責任なやつらの意見に従って，自分のイメージや意志を修正する必要などまったくない。自己満足でもいいから，自分が愛せる作品にしないと，作っていて面白くないし，完成しない。

**2　人の力などあてにするな**

作品制作に入る時点で，何人スタッフがいようとそんなことは関係ない。本気で彼らが本当に手伝ってくれるなんて夢々考えてはいけない。もし，手伝ってくれたらラッキー！　ぐらいに考えよ。いざとなったら（いざとならなくても）自分ひとりでも完成させることが可能である程度の作品を企画すべきだ。

## 次に目標を決める（7月中旬）

作品は目標と締め切り（期限）がないと完成しないものです。皆さんの場合，アマチュアCGAコンテストを目標とされるのが手っとり早いので，年末が締め切りだと思って制作するといいでしょう。しかし，私はさすがにCGAコンテストにはエントリーできませんので，ほかの映像系のコンテストの締め切りを調べてみました。すると，10月末あたりを締め切りにしているコンテストがいくつかありましたので，これを目標にしてみましょう。

次に自分のスケジュールを考えてみました。9月の上旬に10日間ぐらい連続して休みがある。10月にも1週間休みが取れる。9月24日にはサイクロンCG大会があるから，そのとき予告編でも上映して反応を見よう。ということで，以下のような大ざっぱな計画ができました。

～8月末　アイデアを練る
9月上旬　予告編を作る
9月24日　予告編試写会
9月下旬～10月上旬　コツコツ作る
10月中旬　本編作る
10月下旬　仕上げる

こういった計画は，だらだらと遅れていくものですから，最後の締め切りだけでなく，途中で予告編試写会という具体的な締め切りが存在するところがポイントです。なお，この計画は，CGAシステムを十分熟知していて，マシンパワーも十分あるという前提のものですから，皆さんの場合，もう少し余裕を持って，半年ぐらいの制作期間を設けるべきでしょう。

## スタッフを集める（7月末）

“過去の反省”にもあったように，他人はあてにしてはいけません。ですから今回はスタッフは特に集めず，自分ひとりで制作するつもりです。でも，当然プロジェクトルームにはスタッフが常駐していますので，忙しいときはその場にいる人がきっと手伝ってくれるでしょう。

ただ，音楽については，自分で作曲できませんので，いつもどおり小谷君に頼むことにしました。彼は，いままでも，無茶な要求をものともせず，すばらしい曲を数多く提供してくれています。そしてもうひとり，京都芸術大学の1回生の人が参加させてくれといってきました。芸大生だからきっとすばらしいセンスの持ち主でしょう。しかし，彼はCGAシステムを使ったことがない，映像作品も作ったことがない，しかも，当チームのスタッフでもない，などの理由から，とてもじゃないけどあてにはすべきでないでしょう。

スタッフの条件としては，まず，自分と同程度以上CGAシステムを熟知している必要があります。初心者ばかり集めても，指導するのに手間をとってしまい，逆効果です。それから，やはり日頃の実績も大切です。企画の段階ではものすごくやる気を見せても，飽きっぽい性格の人は，肝心なときにいなくなってしまいます。

**リスト1（脚本第1稿）**

**ストーリー**

実は，まだあまり考えていない(これを書きながら考えようか)。とりあえず，予告編を作って，それを見てイメージを固めてからにしようかな?

オープニングは，もうリングの上(短編なんだから，長い前置きなんていらない！)。ファイターはトロル（主人公）とゴブリン。トロルとセコンドの会話で状況説明する。ゴングが鳴って，タイトル。

第1ラウンド。両者一斉にコーナーを飛び出すが，トロルは緊張していて，いきなり自分の足につまずいてこける。いよいよ，頭に血が上り，もう何がなんだかわからなくなって，このラウンドはボコボコに殴られる。

コーナーに戻ったトロルは，「なんでこんなひどい目にあわなきゃいけないんだ！」と嘆く。

セコンド「足を使え，足を！」
トロル「キックは反則でっせ」

さりげなく，ボケる。

セコンド「相手のパンチをよく見ろ。おまえにはよけられるはずだ」という。

第2ラウンドも，開始早々ボコボコに殴られる。

トロル「相手のパンチをよく見ろっていうのは簡単だけど……」相手のパンチをよく見ようとする。「実際は，そう簡単に……」ゴブリンのパンチを初めてかわす。「……できるやん！」

トロルは，少しずつ相手のパンチをよける。よけれるとだんだん落ち着いて，もっとよけれるようになる。そうなると自信も出てくる。相手のパンチをかわして，そのスキに自分のパンチが決まる。

ゴブリンも驚くが，トロルの攻撃がきわめて単調であることに気づき，カウンターを狙う。ゴブリンは，狙いどおりにトロルが打ってきたストレートに合わせて，カウンターを入れる。完全に決まったと思った瞬間，トロルが軽くよける。ここで，第2ラウンド終了。

ゴブリンはトロルがとんでもない才能の持ち主であることに気がつく。このままでは，負けてしまうと直感する。

第3ラウンド。ゴブリンは反則技攻撃に出る。肘打ち，足踏み（相手の足を踏んで，動けなくなったところを襲う）とだんだんエスカレートする。クリンチの状態で，腹からもう1本の腕が伸び，トロルのボディをどついたり，レフリーが，腹を確認しているすきに，ロケットパンチが飛ぶ。

トロルは卑怯者となじるが，ゴブリンはものともせず，絶対に勝ってやるという意志を見せる。

第3ラウンドが終わる。セコンドは，ゴブリンのファイトをむしろ誉める。トロルにこの闘志さえあればと残念がる。自分は，何かの理由でどう頑張ってもファイターにはなれない。それでもあきらめずセコンドをやっている。おまえは才能があるのに情けない。ゴブリンのセコンドになったほうがよかった，と。しかし，トロルは，本当は自分だって勝ちたい。しかし，失敗して，才能がないといわれるのが恐いと本音をもらす。あんまりよく考えていないけど，トロルがやる気を出すきっかけになる。

第4ラウンドは，もう2人の激しいどつき合いになる。ゴブリンもものすごい根性を見せるが，本気を出したトロルとセコンドのコンビの勢いは止められない。ゴブリンは押されだす。

第4ラウンドが終わり，コーナーに戻ったゴブリンは，神に祈るのをやめ，神を呪う。

第5ラウンド開始直後，ゴブリンはマットに倒れる。

選手控え室からゴブリンが出てくる。廊下にトロルが立っているのを見て驚く。敗者を蔑みにきたのかと怒る。トロルは，ゴブリンがこれからどうなるのか心配する。今回は運が悪かっただけ，努力すれば次は私がやられる番だと励ます。

ゴブリン「運？　努力？　フンッ，才能だ！」

ゴブリンは語る。おまえには才能がある。しかしオレはうらやましいなんてこれっぽっちも思わねぇ。オレは，努力して駄目だったから，あきらめられる。あとは，平和に，楽に暮らしていけばいい。だが，おまえは違う。決してあきらめることは許されない。努力を怠ることは許されない。このオレが許しはしない。

トロルは，天から才能を与えられた者の代償は決して安くないことを思い知る。ゴブリンは，ポンとトロルの肩をたたいて，去っていく……。

う～ん，暗いエンディングはいやだなー。晴れ晴れとした雰囲気を出したいなー。

**キャラクター設定**

・ブルートロル

与えられし者。きついトレーニングはいや。やる気も見せない。しかし，本当はファイターになりたいのだが，自分の才能を信じられない。本気を出して負けるようなことになって，自分の才能が偽物といわれるのが恐い。だから，“こんな試合勝っても負けてもいいや”というポーズをしている。

・ レッドゴブリン

与えられなかった者。才能のなさを，陰の努力とハッタリと卑怯なテクニックでカバーしようと必死。本当にファイターになりたいと願っている。

・ドワーフ

トロルのセコンド。子供の頃からファイターに憧れていたが，なんらかの理由で無理だったのでセコンドになった。トロルの才能を見つけ，チャンプに育てようと決意。

## ストーリーを決める（8月）

ストーリーを決めるにあたって，15分以内，制作ができるだけ楽，という2点がポイントとなりました。それから，どうせ作るならコンテストで賞をもらえるぐらいりっぱな作品にしよう！（エライ自信やな） 特に今回は，キャラクター性に重点を置いた作品に挑戦してみたいと思いました。

キャラクター性というのは，要するに登場人物をどれだけ描けるかということでしょう。いままでのCGA作品は，どうもこのキャラクター性が弱いような気がします。好きな作品はいくつもあるのですが，好きな登場人物というのはどうも思い浮かびません（いままでの作品でもっともキャラクターが描けているのは，「ディファイナブル ファンクション」のウサギロボットではないだろうか)。

さて，ストーリーのアイデアは常日頃考えていて，結構たくさんあるのですが，現在のCGではできない，まとまっていない，長すぎるなどの理由で，使えそうなものはほとんどありません。とりあえず，実現可能そうなものを脚本化してみて，ほかのスタッフの意見を聞いてみました(8月上旬)。若い夫婦がある惑星に不時着したあと，無事脱出するまでを描いたファンタジーSFです。しかし，どうみても長すぎるし，主人公たちを表情ゆたかに表現するのが難しいということでボツ！ さらに，別のアイデアを脚本化してみました（8月中旬）が，やっぱり長すぎるということで再びボツ！ とにかくもっと短いものにしなければ！

ちょうどその頃，この連載の都合で人体モデルのサンプルとして，ロボットのボクシングをいろいろ作っていました。基本的なモーションは，ジャブ，ストレート，クラウチ（かがむ）の3つだけなのに，視点をいろいろ変えるだけで，結構「ロッキー」しています。これを使わない手はないと，ボクシングものを作ってみようということになりました。

11月10日，シャープ本社にて，X68000芸術祭近畿地区大会が行われ，当チームからも，マリオネット古本と私がエントリーした。客観的なレポートは，本誌のほかのページで行われていると思うので，エントリー者の視点でのレポートを，裏話を交じえて紹介しよう。でも，会場にOh!X編集部の方は来ていなかったようだが？ まさか，私が行くだろうから，わざわざ大阪まで出張する必要ないと考えたのでは！

*

まず，当チームのエントリー作品だが，当然グラフィック部門で，マリオ古本：監督・原案，かまたゆたか：脚色・制作，CGA界始まって以来のスプラッタホラーCGAだ。タイトルは「EYE」。Graphic Galleryの写真をご覧になれば，雰囲気はつかめると思う。

決して謙遜していうわけではないが，はっきりいって“出来”はよくなかった。本文のGIFTEDの制作もあって，時間がなかったのだ。作品自体は短編なのだが，単なるスプラッタホラーではなく，人間の尊厳に関わる深いテーマを持たせたため，かなり難解で，チーム内の反応もよくなかった。特に我々を傷つけたのは，完成直後に取材に来たおみこしプレスのお姉さま方の感想，「ふ〜ん，なにコレ？」。もう恥をかくだけだから，いまからエントリーを辞退しようとか，どっちみち予選落ちになるから問題ないだろうとか，真剣に考えてしまった。しかし，予選通過通知は来てしまったのだ。

当日エントリー者は，朝10時に集合だ。照明や音響なども交じえて，総合的なリハーサルが行われる。しかし，エントリー者は，ただひたすら待たされているだけだ。もちろん，「呼ばれた方は，この位置まで進んで，マイクを持って，自分の作品の解説を行ってください」という打ち合わせがあったとはいえ，そんなことは，前の人を見ればわかる。なにしろ，予選通過15作品で，我々はエントリーNo.15だったのだ。

マリオ「これはやはり，予選ぎりぎりだったということでは？」

かまた「そんなこと……あるかもしれない」

予定では，予選は10作品だったのだが，近畿地区は応募作品が多く，あまりにレベルが高いので，急遽15作品になったそうだ。そうだとすると，このエントリーNo.はお情けまる出しではないか？（いま考えると，我々は募集の締め切り日の夕方，直接本社まで届けて間に合ったのだから，エントリーNo.が最後なのは当然だ）

さて会場は，大阪の中心からかなり南に外れたところに位置していた。そのため山下氏も来場者が集まるか心配されて，組織票になってもいいから（来場者の投票も審査に大きく影響する)，人を集めてくれと頼まれ，当チームのスタッフなど暇がある者はできるだけ行くようにいっておいた。これで，10数票は有利だ。しかし，私の画策も無駄になる630人という，芸術祭始まって以来の来場者が詰めかけたのだった。

そしてついに，地区大会は始まった。それは，冒頭からきわめてハイレベルな戦いだった。正直にいって，私は芸術祭を少しナメていた。というのも，ほかの地区予選に参加した友人の話などで，ぜんぜんたいしたことないと聞いていたからだ。しかし，それは芸術祭が始まって間もない頃，十分に制作時間がなかった頃の話で，地区大会が進むにつれレベルはどんどん上がっていったのだろう（この調子では，最後の関東地区，九州地区大会は恐ろしいものになっているのでは)。

そんななかで特に目を引いたものをいくつか紹介しよう。まず，京大マイコンクラブ（KMC）がゲーム「RUSH！」を出品していた。ご存じのように，KMCは，“京大クラブは兄弟クラブ”というくだらないシャレが許されるぐらいの内輪だ(このゲームも以前PLAYしたことがある)。このゲーム，ホストとしてPC-9801を使用し，それに4台のX68000をつなげて，8人同時にPLAYできる(ハードも自作)。ルールはシンプルで，自分を相手にぶつけて，床の上から突き飛ばせばいいだけ。それが3Dの画面で，なめらかにアニメーションするわけだが，やりだすと，相当燃えてしまう。自ら縁に立って，突っ込んでくる敵をさっとかわして突き落とすなどの技もいろいろあるし，時間とともに床の一部が崩れていくし，面が進むとそれまでの成績で自分の体重が変更されるなどのゲームバランスまで配慮してある。裏話になるが，このゲームだけは予選をフリーパスしたらしい。なぜなら，予選会場では，PC-9801と4台のXVIをつなげるというような環境はなかったからだ。

ゲームはほかにもいくつか出品されていた。本格的横スクロールシューティングの「XAD LAX」（西村俊也さんほか）は，オーソドックスとはいえ，グラフィックや完成度は完璧であった。なにしろこの作者の方は，いままでもいくつかゲームを発表していて，その腕を見込まれて，タイトーにスカウトされ，就職してしまったというから，ほとんどプロ！

「XAX」は，ゲームとしての完成度は低いものの，1/60秒の速度でフル画面がグルグル回転する技術力は会場に大きなどよめきを起こした。

また，ゲームではないものの，「アフターバーナー」の作者のあの松島徹氏もエントリーされており，ウイニングランのようなポリゴン表示のフライトシミュレータを実に滑らかにアニメーションさせていた。

ミュージック部門も3作品あり，10分にもおよぶ交響曲まで出品されていた。ほかにはツール部門で，超優れもののスプライトエディタ「USEFUL」（荒田降仁さん）があった。実はKMCでも自信のスプライトエディタを出品していたのだが，予選で落選している。1チームからいくつもエントリーするのはいやがられたのだろうと思っていたが，このスプライトエディタを見ると，KMCも納得してしまった。

最後にもうひとつ，今回の大会のレベルを象徴する作品を紹介しよう。宮野元秀さんの「百人一首」だ。これは，百人一首大会に出場する娘さんのために開発されたそうで，ランダムに上の句を読んでくれるものだ。ここまで読んだ方は，「くだらない」と思っただろう（私も最初そう思った)。しかし，この「百人一首」は実に多機能なのだ。万葉集とか，古今和歌集から集められた歌だけを読めとか，女性の歌だけ読めとか，読み人順の何番から何番までを集中的に練習するとか，いろんな読み方がある。百人一首の大会では，“決まり字”とかいうのがあるそうで，“ら”から始まる歌はひとつしかないので，“ら”を聞いたとたんに，取ることができるそうだ。そういった，決まり字の部分しか読まないモード，苦手な歌を選択し，集中的に練習するモードなどもある。つまり，超実戦向け百人一首強化プログラムなのだ。完成度の違いをわかってもらえただろう。

ロボットのボクシングだったら，審判の目を盗んでロケットパンチを飛ばすのはどうだろうかとか，最後両方ともバラバラにふっ飛んでしまうが，たまたま主人公のロボットの足の部品がちゃんと立って，それで勝利をおさめるなんてオチはどうだろうかとか，くだらないアイデアをいろいろ練っていきます。

今回は，キャラクター性にこだわりたいので，単なるギャグには終わらせたくないということで，ぜんぜん別のストーリーのアイデアを強引に取り込んで，タイトルも「GIFTED（与えられし者：才能）」ということに決定しました。でも，シリアスだか，ギャグだかわからない作品になりそう……。

## 脚本から絵コンテまで（8月末から9月初め）

もう制作をあきらめてしまった作品ですから，ストーリーをばらしても問題ないでしょう。リスト1が脚本の第1稿そのものです。脚本，特に第1稿は，書き方やフォーマットなどを気にする必要はありません。簡単に，だいたいどんな話なのかさえわかればいいのです。何人かに読んでもらって，「面白くない」とか「できっこない」とか，いってもらいましょう。

第1稿が，「こんなもんでいいんじゃない」という反応があれば，第2稿を書きます。今回は先に予告編を作ることになっているので，第2稿は予告編の部分だけです。とはいっても，予告編の前半分は，本編のオープニングそのものになっていますので，予告編の制作が無駄になるようなことはありません。

リスト2が第2稿です。第1稿と違い，すべてのシーン，すべての台詞が書き込まれています。そしてリスト3が絵コンテです。しかし，この絵コンテには絵はありません。なぜなら，今回の制作はすべて私ひとりでするつもりで，すべてのカットの絵は頭の中にできているからです。じゃあ，なんのための絵コンテかといえば，カット名を決めるのに必要なのです。

よほど短い作品でもない限り，でたらめにカット名を

---

さて，こうしてつぎつぎと凄い作品が発表されていくと，自然と我々の順番も近くなる。
マリオ「あっ，この作品もすごいですね」
かまた「はははは……」
マリオ「どうします？」
いまさらどうしようもあるものか。今回の応募は，シャープと山下氏に対する義理であって，別に賞を狙ったとかいう大それたものではない。一応予選は通過し，頭数には入ったのだから，それで我々の使命はまっとうされたのだ。そういって自分を納得させるかまたであった。
ついに我々の番が来た。マリオ古本よ，居直るんだ。居直ればなにも恐くはない！
山下氏「どういった作品ですか？」
マリオ「スプラッタホラーです」
かまた「巨大な目玉が出てきますが，彼がなんのために，どこに向かうのか？　それが大きなポイントとなります」
山下氏「それではご覧ください！」
会場が闇に包まれ，静かに「EYE」は始まった。謎の別世界。血のしたたる斧を持ったゴブリン。そして血の池から生まれる巨大な目玉……（以下省略）。
奇跡は起こった。上映が終わるや否や，会場を大きな感動の渦が満たし，割れんばかりの拍手が湧いた（これが大袈裟な表現だと思うなら，誰でもいいからこの会場にいた人に聞いてみるといい。必ず彼は“確かにそのとおりだった”と(笑いながら)答えてくれるだろう）。しかし，これは本当に意外だった。会場の皆さんは，たった一度見ただけで，あの深淵なるテーマを理解することができたのだろうか？　きっとそうだ。そうでなければ，あのような拍手が起こるはずがない。
こうして，我々は予想以上の成果を得ることができたのであった。
このあと，20分間のデモプレイの時間があり，会場の両サイドに用意されたXVIなどで，来場者がエントリー作品を実際にさわってみることができる。我々もメモリを12Mバイトフル実装したマシンでアニメーションさせた。
会場の皆さんの投票がすむと，集計のため審査員の方々は別室にこもり，審査が始まる。この間，会場では最新ゲーム紹介や，ゲームミュージックのライブが行われた。特に例のスターウォーズのオープニングデモ（5分）は感動ものだ。余談になるが，この日の晩，別件でスターウォーズの開発部隊から電話があった。最後の仕上げのために缶詰になっているそうだ。ガンバレ，みんな待っているんだぞ！　しかし，この原稿が本屋に並ぶ頃には，発売されているんだろうか。ちょっと心配。
さて，途中で山下氏が人気投票の中間結果を発表しようとした。
かまた「中間発表というのは，クサイな」
マリオ「どういうこと？」
かまた「だって，最終的な審査結果が出る前に，大賞がわかってしまったら面白くないやん。票数ぐらい操作したはずや」
マリオ「なるほど」
かまた「見てろよ。この中間で，1位を取った作品は絶対大賞取れないで。大賞は，まぁ2位か3位の作品やな」
中間発表の結果，1位は，我々の「EYE」であった……。
いよいよ，審査結果の発表と表彰式が始まった。近畿地区大会では，大賞1名，入選2名が全国大会への出場権を得る。そのほか，協賛各社賞というのがたくさんあって，どうやら，全員何かの賞がいただけるようだ。
我々は，早々と“ニノミヤ賞”が決定した。この瞬間思わずガッツポーズを取った。我々はこの“ニノミヤ賞”を狙っていたのだ。また裏話になるが，会場の舞台や客席の設置は，当チームのスタッフがアルバイトとして動員されていた。そこで，各賞の賞品を展示しながら，品定めをし，ニノミヤ賞のマウンテンバイクが欲しいといって，“淡路号”という名前までつけていたのだ。なにしろ，大賞の賞金が5万円の賞品券で，ほかの賞品は1万円前後のものが多かったから，このニノミヤ賞の大判振舞いは，エントリー者の間から“真の大賞”とまで呼ばれていた（ウソ）。
さて，自分たちの賞が決定すれば気楽なもので，大賞，入選の3つが何になるか，2人で予想をしていた。KMCの「RUSH！」は，身内のひいき目を差し引いて見ても当確だろうというのは一致した。
かまた「ゲームがレベル高いからもう1作入るだろ。でも，どのゲームも，完成度，技術，オリジナリティと甲乙つけがたいな」
マリオ「いや，インパクトという点で弱いですよ。それより，あのスプライトエディタは実力派です」
かまた「でも，松島さんの“S・I・R”もインパクトあるで」
その間にも発表は続けられていくが，我々が本命と見たKMCの「RUSH！」がシステムソフト賞を受賞してしまう。あれっと思っていたら，続けてもうひとつアスキー賞を受賞した。どうやら本当にダブル，トリプル受賞があるらしい。
そしてついに入選の発表。ひとつめは，荒田さんのスプライトエディタ。
マリオ「ほらね」
かまた「じゃぁ，2つめは“S・I・R”か」
続いて2つめの発表。「エントリーNo.15“EYE”」
その瞬間，私は露骨にイヤな顔をしてしまった。というのも，この連載の“はじめに”を読むとわかるのだが，最初の予定では，「GIFTED」が完成しなかったので，この「EYE」の話を書こうと思い，事実，もうかなり書き上げていたのだ。入選して全国大会に出場するのなら，それ以前に内容をすべて掲載するわけにはいかない。原稿の書き直しだぁ！
大賞のほうは，大方の予想どおりKMCの「RUSH！」に決まった。あとで関係者から話を聞くと，会場の人気投票では，「EYE」が勝っていたそうだ。つまりあの中間発表はウソではないらしい。このように，「EYE」が予想外の健闘を見せたのは，発表が最後だったので印象が強かったのと，ゲーム部門，ミュージック部門の作品はレベルの高いものが多く，票が割れてしまったからだと思う。
しかし，この際はっきりといっておく。地区大会にエントリーしたのは我々の責任だが，全国大会に行くはめになったのは，来場者みんなの責任だぞ。我々の受賞インタビューはひと言「君たち，本当にいいのか?!」（この言葉の意味は，全国大会で明らかになるであろう）。
以上のように近畿地区大会は大きく盛り上がり，我々も存分に楽しめ，大成功といってよいだろう。ここまで読んでくれた読者はおわかりいただけたと思うが，“踊るアホウに，見るアホウ”というとおり，こういったイベントは，ただ見にいくより，エントリーしたほうがずっと面白い。芸術祭は，来年以降も続けていくそうなので，ぜひ皆さんも積極的に参加しよう。
それでは，来年の6月号で，芸術祭全国大会スペシャルレポート「感動の神風は2度吹くか!?」をお楽しみに。
P.S.　その夜，日本橋筋を，ニノミヤ賞ののし紙のついたマウンテンバイクで帰る男の姿があった。

つけているとすぐわけがわからなくなります。実際の作品制作では，レンダリングした画像に対して，オーバーラップや，お絵描きツールによる書き込み，合成などを行いますので，フレームファイル名がそのまま画像ファイル名にならないケースが非常に多く発生します。それらをちゃんと管理しておかないと，タイムチャートの作成が大変ですし，後日手直しを加えるときも困るし，制作の最終段階で，完成しているカットと完成していないものがわからなくなって，もうパニックを引き起こします(経験者は語る)。カット名は，計画的，系統だってつけることを強くお勧めします。つまり，この絵コンテは，どんな絵かを解説するものではなく，作品を作るための設計図だと考えるべきです。一見ただの1カットに思えても，フェードインするのであれば，その部分は違う名前をつけるべきだし，セリフ（今回はすべてテロップ）が入ると合成が1回増えるので，もう別のカットです。ですから，慎重に，綿密に組み立てていきます。

しかしながら，制作中にさまざまな変更が発生するのは避けられません。そうすると，“このカットの出だしはフェードイン”“このカット削除”“ここにセリフ追加”とかが鉛筆で書き加えられ，だんだん見にくいものになってしまいます（リスト3は，そういった変更を直したもの）。その点，絵コンテをワープロなどで作っておけば，比較的修正は楽になるでしょう。

**リスト2（予告編第2稿）**

ブルートロルは，左足の貧乏揺すりをしている。右ひざに手をやったが，止まらなかった。そんなトロルの肩に，セコンドのドワーフが手をかけた。
ドワーフ「恐いか？　初めての試合だからな。だが，それはヤツだって同じだ」
　ドワーフの後ろでは，レフリーに呼ばれた対戦相手が観客に紹介されていた。
レフリー「赤コーナー，450馬力，BWW製，レッドゴブリン」
　トロルもゆっくりとリング中央に進む。
レフリー「続きまして，青コーナー，435馬力，シャムソン製，ブルートロル」
　トロルはおどおどと，片手を上げる。そんな，トロルをゴブリンがにらんでいる。
ゴブリン「オレと同じリングに上がったことを後悔させてやるぜ。ゴングが鳴る前にさっさと農場に帰っちまいな」
　コーナーに戻ってきたトロルは何か考えている。
ドワーフ「どうした？」
トロル　「いや，農場へ帰ろうかなって」
ドワーフ「バカやろう。しっかりしろ」
　ドワーフはトロルをイスに座らせる。
ドワーフ「おまえ，ファイターになれなきゃ，農場どころか，土方か，へたすりゃスクラップってわかってんのか」
　ドワーフがトロルをにらむ。
ドワーフ「おまえには才能がある。あとは，おまえのやる気だけだ」
　レフリーがドワーフに，リングから降りるように指示する。
ドワーフ「自分の道を切り開いてこい！」
　ドワーフがパシッとトロルの背中を叩いた。
　試合開始のゴングが鳴った。

タイトル　GIFTED

テロップ　神は，平等ではない
　God is not equality.

　デモ用に作ったボクシングシーンを数カット挿入。

テロップ　与えられし者と与えられない者
　Gifted, not Gifted.

ゴブリンのアップ
ゴブリン「運？　努力？　フン！　才能だ！」

テロップ　勝つためには手段を選ばない男　レッドゴブリン

ゴブリンがこっちを向いて，「ロケットパーンチ！」

トロルのアップ
トロル　「なんでこんなひどい目にあわなきゃいけないんだ！」

テロップ　未完の大器　ブルートロル

トロルの目が光る。「いまだ！」猛然とゴブリンにぶつかっていく。
トロル　「ロイヤル・サンダー・トルネード，実はただのアッパー!!!」
　ゴブリンに軽くよけられ，スカッと空振りする。

テロップ　勝者は一人，そして敗者も一人
　only one victor,and defeated

　再びゴングの音。両者いっせいにコーナーを飛び出していく。両者が同時に最初のパンチを出す。

タイトル　GIFTED　　制作好調

## 形状データを作る

通常の作品では，この作業がかなりかかります。特に予想外の手間がかかるのが，背景の作成です。背景は，シーンごとに変わるし，カメラが動くとかなり広範囲が見えてしまうし，かなり遠くのほうまで作らないといけません。努力して細かく作ったからといって目立つわけでもないくせに，手を抜くと画面全体が安っぽく見えます。

今回は，ラストシーンを除いてすべてリング内で，そのリングもデモ制作時に作ってあります。本当は，リングの周りの客席や観客を作るべきでしょうが，明るいリングから見れば，周りはほとんど見えないだろうということで，背景はまったく何も置きませんでした。ちょっと強引でしたが，結果的には，ほとんど問題ありません。むしろ，ちゃちな人型でも並べたほうが不自然になったでしょう。ただ，極端なあおりのカットのとき，“あしたのジョー”みたいな感じで，背後の天井のライトがギラギラ輝いているという効果はやってみたかったので，天井のライトだけは制作しました。

さて，そうすると問題は，レフリーとセコンドの人体モデルだけです。しかし，これをまともに作っていたらとても大変です。私はさっさとあきらめ，レフリーはスタッフの砂川君が別の作品用に制作したデータ（モビルスーツの整備士）を，セコンドは，「SWORD」の森山氏が制作して送ってきてくださったデータをそのまま流用することにしました。すごい手抜きですね。プライドも何もあったもんじゃない。

## いよいよ制作に入る（9月）

ここまでは平和そのものです。上記の文章にも“8月上旬”なんて時間に余裕のある書き方をしていますが，実際に制作に入ると，スケジュールは，日単位，最終的には時間単位になっていきます。それと同時にありとあらゆるトラブルが発生し，だんだんパニック状態になっていくのです。ここまでが，天国企画編だったのに対して，ここからは激闘制作編だ！（文調も変わる）

9月6日，いよいよクランクインだ。ところが，いきなりこの日から，プロジェクトルームの模様替えが始まった。ナンテコッタイ！　結局まる2日間つぶれてしまった。予定外の仕事が発生するのはよくあることだ。部屋がきれいになって，作品制作も効率がいいとでも考えてあきらめよう。

8日，まずは，24日までに予告編を作ればいい。ややこしいことは考えずに前から順番に作っていこう。なになに，カットAはトロルが貧乏揺すりをしていて，その

ひざに手をやるところだな。トロルもリングも形状データはできているし，トロルは，標準人体モデルですべて関数化されているし，楽勝楽勝……。

## 1カットができない！

まずは，貧乏揺すりのモーションファイルを制作するところから始まった。その頃は，まだMFE(モーションファイル エディタ：タケルのロボットデータ集についているツール)ができていなかったので，エディタで各関節のデータを入れ，試行錯誤を繰り返すしかなかった。もちろん，こういった試行錯誤は，トロルのデータではなく，非常に面数の少ないBOX君を使って行っている。それでもなかなかうまくいかない。

まず，構図がひざを中心とする足のかなりのアップなので，パッと見て，なんの絵かわからない。それに，貧乏揺すりというより，体がケイレンを起こしているように見える。カメラの位置をいろいろ変えてみて，足全体が映るようにする。そうすると，もう片方の足が入ってきて，ひざを隠してしまう。なにしろ，FFEが構造体に対応していないため，構図の確認はいちいちレンダリングするので非常に時間がかかる。このカットだけ，もう片方の足は削ってしまおうかとも考えたが，何度かやり直しているうちに，なんとか足に見えるような位置を見つけることができた。

貧乏揺すりのほうも，周期や幅を変えてみる。やっぱり，貧乏揺すりには見えない。なぜ見えないのだろう。ケイレンと貧乏揺すりはどこが違うのだ。自分で貧乏揺すりをやってみてじっと観察する。すると，やっと違いがわかった。ケイレンは，体が中心になって動くが，貧乏揺すりは，爪先が中心になって動く。つまり，爪先がまったく動かないようにすればいいのだ。人体モデルは腰が中心になっているので，爪先を固定してひざを動かすというのは，かなり難しいのだが，原因がわかれば，数回の試行錯誤でそれらしいモーションになった。

## やっぱり1カットができない！

BOX君の代わりにトロルを使い，リングの上に並べてみる。そこで，トロルはリングサイドでイスに座っているので，イスの形状データを作らないといけないことに気がつく。木でできた安物のイスというイメージだが，質感を出すのが結構苦労した。さっそくイスに座らせようとしたが，うまくトロルのお尻がイスの上の面にくっつかない。イスのスケールを変更すると，縦横比や，トロルとのバランスがとれない。イスのデザインを変え，もう一度作り直すが，それでも，少しイスにお尻が食い込んでしまう。結局，きりがないので，もう妥協することにした。

そして，再びすべてをリングに並べ，カメラを設置し，レンダリングしてみたら，ショック！　カメラの位置とリングのロープの位置が一致してしまい，画面にロープが食い込んだような画像ができてしまった。カメラを前にずらすと，また足がアップになりすぎてしまう。これは，画角を広げるという手段で回避した。

**リスト3（予告編絵コンテ）**

```
DōGA   フライング ロゴ

TEL0   オープニング　クレジット（1枚）

BL     ブラック
A1F    フェードイン
A1     貧乏揺すりをしているトロルの足のアップ
A1     一度貧乏揺すりが止まる
A1     再び貧乏揺すりが始まる
A2     カメラパンして，トロルの顔へ　あおり
A2T1   ドワーフ「恐いか?」
A3     トロルの視線が自分のひざから正面へ。ちょっと止める
A4     トロルの視線が，落ちる
A4T1   トロル「ちょっとな……」
B1T1   トロル「ちょっとな……」（せりふの途中でカット変える）
B1     トロルとドワーフが向かい合っている
B1T2   ドワーフ「初めての試合だからな」
       ドワーフ，セリフと同時にあごが動く
C1T1   ドワーフ「だが それは」
C1     ドワーフのアップ。ドワーフの肩ごしに，リング中央，ゴブリンとレフリーが
       見えている
C1T2   ドワーフ「ヤツだって同じだ」
C1     ドワーフ，振り返り気味に，右手の親指で後ろを指す

       ==　BGMスタート　==

TEL1   テロップ　　　　神は　平等ではない
       （テロップは，上下に分かれている状態からくっつく）
TEL2   テロップ　　　　与えられし者　与えられなかった者

D1     レフリーとゴブリン，リング中央，並んで立っている。
       ややあおりカメラはレフリーからゆっくり横に移動し，ゴブリンに移っていく
D1T1   レフリー「赤コーナー」
D1T2           「450馬力」
D1T3           「BWW製」
D2     レフリー片手を挙げる
D2T1   レフリー「レッド ゴブリン」
D3     ゴブリン，胸を張り，両手を振り上げる
E1     トロルからの視点。リング中央へ。歩調に合わせてカメラ上下
E1T1   レフリー「続きまして」
E2T1           「青コーナー」
       ゴブリン，こっちを向く
F1T1   「青コーナー」（セリフの途中でカット変える）
F1     トロル歩いているところ。横から
G1T1   レフリー「435馬力」
       カメラ，リング中央，正面。トロルがレフリーの横にくる
G2     トロルが停止し，正面を向く
G3     レフリーが片手を上げる
G3T1   レフリー「シャムソン製」
G3T2           「ブルー トロル」
G4     トロル，片手を上げる
H1     ゴブリン画面いっぱいのアップ
H1T1   ゴブリン「オイ！」
I1     ゴブリンとトロルが並んでいる。ゴブリンにらんでいる
I1T1   ゴブリン「同じリングに上がった」
I1T2           「ことを後悔させてやる」
I1T3           「さっさと農場に」
I1T4           「帰っちまいな」
I2     ゴブリン，肩でトロルを押す。トロル引く
J1     トロル，リングサイドへ。ドワーフ，トロルを見ながら
J1T1   ドワーフ「どうした?」
J1     トロル，歩くのをやめる
J2     トロル，ロープに手を伸ばす
J3     トロル，ロープをつかむ。視線を上げる
J3T1   トロル　「農場へ帰ろうかな」

TEL3   テロップ　　　　勝者は一人　敗者も一人
TEL4   テロップ　　　　努力，運，　そして……才能

K1     ドワーフ，バストショット。あおり
       カメラはゆっくりと横へ移動
K1T1   ドワーフ「バカやろう！」
K1T2           「農場どころか」
K1T3           「へたすりゃ」
K1T4           「スクラップだぞ！」
L1     同様にトロルのバストショット。カメラ，逆方向に移動
       トロル視線を落としている
L1T1   ドワーフ「おまえには才能がある」
L1T2           「あとは，やる気だけだ！」
L2     トロルの視線がゆっくりと上がっていく
M1     レフリー，両手を広げる。超あおり，以後1カット短く
M1T1   レフリー「セコンド アウト！」
N1     トロル，自分の拳を合わせる
O1     トロルの顔のアップ。視線が下から正面へ，ズームアップ
O1T1   トロル「いくぞ!!」
P1     ゴングのアップ。ゴングの音

TIT    タイトル　GIFTED
TEL5   テロップ　制作好調
```

## それでも1カットができない！

いろいろカメラの位置を修正している途中で，誤ってリングから大きく外れたところに設定してしまった。それで気がついたのだが，トロルの位置がリングの端すぎて，肩の部分がロープに食い込んでいる。しかし，このように1カットの制作に手間取っているとだんだんいやになってくるので，もうどうせ本来のカメラの位置からは見えないのだからとさっさと妥協した。このように，なんでも妥協してしまう状態を“通しモード”と呼ぶ。

さて，次にひざに手をやるモーションだが，貧乏揺すりと同様BOX君でまず作ってみる。これは思ったより簡単に（偶然に）できた。しかし，それをトロルに当てはめると，手の位置とひざの位置が大きくずれてしまう。もともとこのトロルは，手足のバランスが通常の人体と大きく異なる場合でも有効かどうかを実験するために，異様に上半身を大きくしてあったのだ。こうなると，もうBOX君でのテストは全然使いものにならず，トロル自身で試行錯誤するしかない。でも，どうやってもひざに手がこない。どうしたらいいんだ！

今後のことを考えて，トロルのデザインを一から作り直そうか。でもそれだけで数日はかかる。えぇい！　こんなカットは省略だ！　一度，貧乏揺すりを止めて，ちょっと間をおき，無意識に再び貧乏揺すりを始めるだけで，十分トロルの不安感を表現できるではないか。もう，ややこしいことはほっといてどんどん制作を進めるぞ！

しかし，そのとき京都芸大の知人から連絡があり，学校の課題のためどうしても今日，明日の2日間，プロジェクトルームのすべての機材を貸し切りたいという。おいっ！　こっちもせっぱ詰まっているんだぞ。いったいどうなるんだーーー！

予告編上映日まで，あと　13日
完成したカット　　　　　0

## おわりに

今回は，本文が少ないような気もしますが，芸術祭のレポートでも読んで納得してください。次号は，真の最終回「山越え，谷越え，どこまでも（ひたすら激闘編）」と連載のまとめ，「CGAと日本文化（仮称）」をお届けします（今度こそ真の最終回だ！）。

さて年末年始，皆さんいかがお過ごしですか？　我々は恒例の“なんでも鍋”などつつきながら，CGAシステムのバージョンアップの準備などで泊まり込みです。冬休みということで，プロジェクトルームに遊びに来る方もいらっしゃるでしょう。そのときは，いっしょに鍋を囲みましょうね。ただし材料は持参ですよ。

# 各読者連絡事項

**・CGAシステムを持っていない人も遊べるロボットデータ集ついに登場!**

ずいぶん遅れてゴメン。もう何回もアナウンスしたような気もするけど，標準人体モデルのデータ集が，なんとか完成致しました。もっと形状データを増やそうとか，もっと凝った動きを目指していたら，いつまでたっても完成しなかったのだが，ついにディスクがいっぱいになってしまった。

このデータ集には，単純なテスト用人体モデル（BOX君）から超リアル人体モデルのほか，モンスターやロボットなどの10体の形状データと，歩く，走る，ジャブ，ストレート，チョップなどの10種類のモーションデータが入っている。ということは，10×10＝100通りのアニメーションが作れる……フッ，甘いな。視点の位置など自由に変更できるので，組み合わせは無限だ。

それに，“データ集”とあるが，実際にはデータよりプログラムのほうが多い。作画ソフトやアニメーション再生ソフトのほか，全自動CGアニメーション制作システム（仮称）まで入っている。だから，メニューから好きな形状データとモーションデータをマウスで選択して実行ボタンを押すだけで，作画，アニメーションのすべてを自動的にやってくれる。CGAシステムを持っていなくても，CGの難しい知識など何も知らなくても，誰にでもアニメーションができる！（スタッフ内では，ネコでもできるといわれている）

作画したデータは，別のディスクに蓄積され，同時にアニメーションのシステムも自動的にコピーされるので，そのディスクだけで独立したCGAデモディスクとなる。友人に見せるのもよし，学祭や新入生募集のデモにでもすれば効果はバッチリだ。

さらに，新しいツールとして，標準人体モデル用のモーションエディタもついているので，腕に覚えのある人は，前回の連載を見ながら，オリジナルのモーションを作ったり，複数のモデルを登場させて，ストーリー作品へと応用していくことも可能だ。

今回のデータ集だが，個人的にはCGAシステムを持っていない人に使ってもらいたいな。自分のマシンのディスプレイで，アニメーションが見られる感動は，実際にやってみなければ決してわからないから（感動したからって，すぐにCGAシステムを申し込むなよ。前回，“めんどくさいCGA配布係”を公開したら，ドバーと応募があって困ってるからな。もうすぐバージョンアップするっていってるんだから，もう少し待ってよ。ごめんね）。

さて，このデータ集の入手方法だが，例によって，ソフト自動販売機タケルで買ってください。値段は，当チームとしてはタダにしたいけど，タケルの使用料として1,000円にはなる。そのくらいいいでしょ？　直接こちらに申し込んでもらっても，発送の手間や手数料を考えると同じぐらいになっちゃうからね。この号が出る頃には，入手できるはずだから，まぁよろしくね。

**・第4回　アマチュアCGAコンテスト**

えーと，皆さんご存じだと思いますけど，CGAコンテストの締め切りは，12月31日です。今年は，例年多数の作品を応募してくる京大マイコンクラブが1作しか制作していないし，昨年のグランプリ授賞者の森山氏も忙しくて制作を断念しているとかで，応募数が少ないんじゃないかと心配しています。

例年ものすごい作品ばかりが集まっていますが，ほんの短い作品でもいいですよ。どこぞのなんとかいう作品のように，一発ギャグでもいいですよ。CGAシステムを使っていなくても，全然関係ないですよ。

いまから，応募用紙を取り寄せていると間に合わなくなりますから，とりあえず作品を送ってください。応募先は，

〒533 大阪市東淀川区淡路5-17-2 102号
CGAコンテスト事務局

です。なお，BGMやキャラクターデザインなど，著作権には十分ご注意ください（ダークサイドはなしですよ）。

**・オリジナル曲ご提供のお願い**

毎年，CGAコンテストを主催するたびに問題になるのが，BGMの著作権問題です。

一般的にいって，CGアニメーションを制作する方が，曲もオリジナルに制作できるとは限りません。たいていの場合，既存の曲を使って，著作権協会への手続きに苦労する（場合によっては，苦労してもクリアされない）ケースがほとんどです。

このように，CGアニメーションを普及させていくうえで，BGMは大きな問題になります。

そこで，ぜひ皆さんにお願いがあります。皆さん，あるいはその友人で作曲ができる方，どうか著作権フリーの曲をご提供いただけないでしょうか？

皆さんから集めました曲は，著作権フリー曲集として，CGAコンテスト応募者のほか，CGA制作者に広く公開し，ご利用いただけるようにしていきたいと思います。

曲のフォーマットは，OPMファイルほか，なんでも結構です。詳しくは上記「CGAコンテスト事務局」までお問い合わせください。

X68000・Z-MUSIC用 

# DRAGON SABERより 4面 地葦

Shindo Noriyuki
進藤 慶到

X68000・OPMD用

# すき

Sakai Tohru
酒井 徹

X1・MusicBASIC用

# THE ENTERTAINER

Uehara Hiroshi
上原 寛

年が明けてLIVE inも'92に。皆さん，いかがお過ごしですか？ 今月は，ちょっとしたお年玉気分で，いろいろな方面の曲を集めてみました。ゲームミュージック，邦楽ポップス，スタンダードと，まさにごちゃまぜ福袋。ぜひ打ち込んでくださいね。

DRAGON SABER

## こんにちは，進藤です

あけましておめでとうございます（一度雑誌でこれをやってみたかった)。

さて，12月号は音楽特集ということで2曲のサンプルを選んでみたわけですが，どうでしたか？ 感想などを聞かせてもらえれば嬉しいです。ただ，2曲ともBASICリストではなかったので，なかには戸惑った方もいるかもしれませんね（少々説明不足だったし。そういえば，あのリストは行番号をあとから追加して印字したものなので，TABで桁をそろえたはずなのにズレてしまったところがある)。

今月の曲はBASICで書いてます。一応BASICでもZ-MUSICの動作確認（正確にはMUSICZ.FNCのバグ出し）をしたので，その集大成といった感じで作りました。曲はナムコの「DRAGON SABER」より，「4面 地葦」です。難しい曲なので作る気はまったくなかったのですが，とある筋からの熱烈なリクエストに負けてしまいました。トホホ。

ゲーム自体はちょっとバイオな（?）シューティングといったところでしょうか。ストーリー的にはあの「DRAGON SPIRIT」の続編になってまして，X68000への移植を望む声も結構多いようです。どちらかというとマニア向けで大ヒットには至りませんでしたが，例によってシステム2によるサウンドは絶品です。音源の性能を使い切っていて，なおかつ洗練されています。CDはおすすめですよ。

プログラムについてですが，演奏にはZ-MUSICは当然として，ほかにCM-64が必要となります。内蔵，MIDI合わせて22トラック使用しています。これはやたら多いように感じられますが，FM&AD PCMでだいたいのイメージを演奏させていて，MIDIのほうはチョッパーベース，シンバル，オルガン，ストリングス，あとはほとんど音を厚くするために重ねているだけなので，実際に鳴っている音はそんなに多くないです。ですから，MIDIを装備していない人もFM&AD PCMトラックを打ち込めばある程度の演奏は可能です。また，ほかのMIDI楽器への対応もそれほど苦労しないでしょう（SC-55なんかいいかも)。

制作は苦労続きでした。FMの音色はいくらいじっても全然似た音ができないし，曲の後半はなかなか音取りが進まなかったりLAがちゃんと鳴ってくれなかったりして大変でした。ハイハット&シンバルを再現するのもひと苦労。完成までにはおよそ1カ月くらいかかってしまいました（その間ずっとこの曲を聴いていたわけで，さすがに耳にタコができてしまいましたよ私は)。音楽を真面目にやってこなかったツケですかね。ハハハ。

というわけで，打ち込んで楽しんでいただければ幸いです。できたら，こちらのほうも感想などいただけたら嬉しいな。それでは。 （進藤慶到）

## いわれてみたい

X68000のOPMD用には渡辺美里さんの「すき」をお送りしましょう。渡辺美里さんはCMソングなどでも有名ですよね。確か「日本生命のロングラン」だったと思いますが，ズバリ，この曲もCMソングになっています。「カルタス千里走る」でお馴染みの（?）大江千里が，渡辺美里のために作曲をした曲です。

彼女はこのLIVE inでは過去何回か掲載されていて，「LONG NIGHT」，「恋したっていいじゃない」に続き，「すき」は3回目ということになります（こうやって書くと，なんだか紅白歌合戦みたいだ)。

渡辺美里

投稿してきてくれたのは，1991年の7月号でリンドバーグの「今すぐKISS ME」が掲載されている酒井君です。

さて，作品です。前回ではZMUSIC.FNCを使用していたようですが，今回はYコマンドをなるべく使わない曲を選んでいることもあって，ZMUSIC.FNCは使っていません。選曲がうまくいっていることもあり，ZMUSIC.FNCを使わなくても十分に聴きごたえのある作品に仕上がっています。

メロディやヴォーカルの音色もイメージ的に合格点が与えられます。作品のデキもかなりよいでしょう。

この作品ではパーカッション系はサンプリング主体ではなく，FM音源がかなり鳴っています。もともとFM音源の得意分野ではなく，どちらかといえば不得意分野に入る系列の音ですので，音色はかなり気合を入れてもちょっと貧弱に聞こえてしまい

ますよね。そこで，もっとボリュームを下げてみてはどうでしょう。1660行でV15としているのをV11くらいに抑えてみると，サンプリングが前面に出てきて引き締まった感じになります。いかがですか？

酒井君は受験生。これから大切な時期にさしかかります。数ある作品たちのなかから掲載されるわけですから，実力だけでなく運も味方しているのでしょう。無理をしない程度に無理をして（？）身体に気をつけてがんばってくださいね。朗報をお待ちしてます。

## 誰もが知ってるあの曲を

X 1のMusicBASIC用には，洋画「スティング」にもアレンジして使われていた「THE ENTERTAINER」をおとどけします。この曲はあまりにもポピュラーですよね。タイトルを知らない人でも聴いたことはあるはずです。試しに入力してみてください。ほら，知ってる曲だ。

音色はホンキートンクピアノだけなのにとっても味わいがあります。

リストは非常に短いですよね。もちろん，音色データがほとんどないことも影響してはいますが，短くしようとする努力もあちらこちらに見受けられます。やっぱり入力する人のことも考えて作りたいですよね。

この作品では，ちょっと変わったテクニックが披露されています。

PLAY ”［t70r16 t140r16］”

というものです。

本来のテンポをAで演奏しているときに，

B=3/4＊A

C=6/4＊A

PLAY ”［tBr16 tCr16］”

とすると，16分音符以下の音は「音符・休符・音符」の3連符になります。8分音符以上の音符はテンポAになります。わかる人には説明するまでもないでしょう。ちなみにB=70, C=140ということで，この曲はテンポ93くらいで演奏されていることになります。

意味のわからない人は以下のプログラムで試してみてください。

PLAY ”L16 t120 AAAA AAAA”

PLAY ”L16 AAAA AAAA:［t180r16 t90r16］”

上と下を聴き比べると明らかに異なるはずなのに，演奏時間は同じになると思います。さらにTEMPO 0を実行したのち，両方の最初にある ”L16” を ”L8” に変えてみてください。今度は両方とも同じように聞こえたはずでです。

さて，作品のほうに話を戻しましょう。最初に［4/4］か［24/16］と聞いてきます。前者は普通に演奏して，後者は前記のテクニックを使った演奏になっています。まさに，1粒で二度おいしい構成にですね。

実は，同封のT-SQUAREの「IT'S MAGIC」のほうが曲の完成度，洗練度ともに上をいっていたのですが，曲の短さ（身近さとも書く）や楽しさで今回は「THE ENTERTAINER」を選びました。まあ，「IT'S MAGIC」はいずれ登場してもらうことになるでしょう。本当に凄いんだぞ。

久し振りに期待の大型新人，今後の動向が気にかかります。上原君，ちゃんと投稿するんだぞ。それではまた来月。　(S.K.)

図1　DRAGON SABER用カウンタ表示

```
-- counter --
 1:00000048 00001E00   2:00000054 00001E00   3:0000006C 00001E00   4:00000048 00001E00
 5:00000048 00001E00   6:00000048 00001E00   7:00000048 00001E00   8:00000048 00001E00
 9:00000168 00001E00  10:00000000 00000000  11:000001C8 00001E00  12:00000048 00001E00
13:00000048 00001E00  14:0000004A 00001E00  15:0000004A 00001E00  16:0000004A 00001E00
17:00000000 00000000  18:00000000 00000000  19:00000168 00001E00  20:00000048 00001E00
21:00000090 00001E00  22:00000048 00001E00  23:00000079 00001E00  24:00000079 00001E00
25:00000079 00001E00
```

### リスト1　DRAGON SABER

```
  10 /*    save "DSA4                    .bas"
  20 /*
  30 /*    DRAGON SABER
  40 /*
  50 /*        4面 地 寶
  60 /*
  70 /*       (C) NAMCO
  80 /*
  90 /* PROGRAMED BY ENG
 100 /*
 110 /* - X68000 & CM64  for ZMUSIC.X -
 120 /*
 130 m_init()
 140 m_ch("fm")
 150 m_adpcm_block("dsa4.zpd")
 160 /*
 210 key  3,"         @M
 220 key  9,"m_stop()@M
 230 key 10,"m_play()@M
 290 /*
 300 char o(255),v(4,10),pa(6)
 310 str p(30)[256]
 390 /*
 400 for i=1 to 25
 410  m_alloc(i,2000)
 420  m_assign(i,i)
 430 next
 490 /*
 500 VD()       /* FM音源データセット
 510 CM()       /* CM64初期設定
 520 MUS()      /* MMLデータ定義
 560 m_play()
 570 end
1000 /*
1010 /* SET MML TO TRACK
1020 /*
1030 func t(tt)
1040  r=0
1050  while o(r)<>255
1060   m_trk(tt,p(o(r)))
1070   r=r+1
1080  endwhile
1090 endfunc
2000 /*
2010 /* CM64 SETUP
2020 /*
2030 /* ここで行っている設定はTWIN1・ZMSと
2040 /* ほとんど同じですので参考して下さい
2050 /*
2100 func CM()
2110 o={&H7F,0,0,0}:m_roland(16,22,o,3) /* 初期化
2120 o={     16, 0, 1,                  /* Address (LA)
2130          0, 4, 4,                  /* Reverb
2140          2, 4, 4, 6, 6, 6, 0, 0, 4, /* Ptl reserve
2150          1, 2, 3, 4, 5, 6, 7, 8, 9} /* MIDI ch#
2160  m_roland(16,22,o,24)
2170 o={     82, 0, 1,                  /* Address (PCM)
2180          2, 4, 4,                  /* Reverb
2190          6, 5, 5, 5, 5, 5,         /* Ptl reserve
2200         10,11,12,13,14,15}         /* MIDI ch#
2210  m_roland(16,22,o,18)
2220 /*
2230 /* LA SOUND SET
2240 /*（楽器のメモリに音色をセットする）
2250 /*
2260 /* 面倒臭いので M_ROLAND() 命令を使いましたが、もちろん
2270 /* MT32_COMMON()&MT32_PARTIAL()でも設定出来ます(手抜き)
2280 /*
2290 /*  CRASH SYMBAL
2300 /*
3000 o={   8,  0,  0,
3010  67,114, 97,115,104, 49, 32, 32, 32, 32,
3020   2,  0,  1,  0,
3030  36, 50, 11,  0,  0,  7, 50,  7,
3040   5,  0,  0, 50, 50, 50, 50, 50, 50, 50, 50, 50,
3050  62, 20, 50,
3060  50,  0, 11,  0,  7,  0,
3070   0,  0,  0,  0, 50, 50, 50, 50,100,100,100,100,
3080 100, 50, 91, 12, 27, 12,
3090   0,  0,  5, 43, 47, 80, 80, 85, 77, 59,  0}
3100  m_roland(16,22,o,75)
3110 /*
3120 /*  CLOSED HIHAT
3130 /*
3200 o={   8,  2,  0,
3210  72,105,104, 97,116, 32, 32, 32, 32, 32,
3220   2,  0,  1,  0,
3230  36, 50, 11,  0,  0,  4, 50,  7,
3240   5,  0,  0, 50, 50, 50, 50, 50, 50, 50, 50, 50,
3250  62, 20, 50,
3260  50,  0, 10,  0,  7,  0,
3270   0,  0,  0,  0, 50, 50, 50, 50,100,100,100,100,
3280 100, 50, 91, 12, 27, 12,
3290   0,  0,  0, 33, 20, 28, 12,100, 83, 77, 31}
3300  m_roland(16,22,o,75)
3310 /*
3320 /*  OPEN HIHAT
3330 /*
3400 o={   8,  4,  0,
3410  79,112,101,110,104, 97,116, 32, 32, 32,
```

```
3420  2, 0, 1, 0,
3430 36, 50, 11, 0, 0, 5, 50, 7,
3440  5, 0, 0, 50, 50, 50, 50, 50, 50, 50, 50, 50,
3450 62, 20, 50,
3460 50, 0, 10, 0, 7, 0,
3470  0, 0, 0, 0, 50, 50, 50, 50,100,100,100,100,
3480 100, 50, 91, 12, 27, 12,
3490  0, 0, 0, 2, 54, 0, 12,100, 89, 47, 0}
3500 m_roland(16,22,o,75)
3510 /*
3520 /* DSA SYNTH2
3530 /*
3600 o={ 8, 8, 0,
3610 68, 83, 65, 99,104,111,114,100, 32, 32,
3620  2, 5, 3, 0,
3630 36, 55, 16, 1, 0, 37, 88, 7,
3640  0, 0, 0, 0, 0, 0, 0, 50, 50, 50, 50, 62,
3650 65, 19, 60,
3660 99, 15, 9,105, 3, 70,
3670  0, 0, 0, 0, 32, 72, 80,100, 84, 46, 14, 0,
3680 99, 60, 91, 12, 27, 12,
3690  3, 4, 0, 15, 82, 78, 71,100, 82, 60, 0,
3700 36, 45, 16, 1, 0, 0, 88, 7,
3710  0, 0, 0, 0, 0, 0, 0, 50, 50, 50, 50, 62,
3720 65, 19, 65,
3730 74, 18, 13,105, 3, 70,
3740  0, 0, 0, 0, 32, 72, 80,100, 84, 46, 14, 0,
3750 99, 85, 91, 12, 27, 12,
3760  3, 4, 0, 15, 82, 78, 71,100, 82, 60, 0}
3770 m_roland(16,22,o,133)
4940 /*
4950 /* パッチパラメータを変更する
4960 /* (LAの音色番号 126~128,2)
4970 /* これで初めて楽器のメモリに登録した音色が
4980 /* 使えるようになります
4990 /*
5000 pa={ 2, 0,24,50, 0, 0, 1} : mt32_patch(126,pa,16)
5010 pa={ 2, 1,24,50, 0, 0, 1} : mt32_patch(127,pa,16)
5020 pa={ 2, 2,24,50, 0, 0, 1} : mt32_patch(128,pa,16)
5030 pa={ 2, 4,24,45,12, 1, 1} : mt32_patch( 2,pa,16)
5990 endfunc
6000 /*
6010 /* FM VOICE DATA
6020 /*
6030 func VD()
6040 /*
7000 /*  AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2 AME  BASS
7010 v={ 31, 15,  4, 15,  3, 23,  3,  0,  0,  0,  0,
7020     31, 15,  0, 15,  2, 45,  2, 11,  3,  0,  0,
7030     31, 17,  0, 15,  2, 25,  2,  0,  0,  0,  0,
7040     31,  0, 11, 15,  0,  0,  1,  1,  0,  0,  0,
7050 /* CON FBL  SM PAN
7060     2,  2, 15,  3}
7070 m_fmvset(70,v)
7080 /*
7100 /*  AR D1R D2R  RR D1L  TL  RS MUL DT1 DT2 AME  SOLO
7110 v={ 21, 12,  0, 15,  1, 20,  0,  3,  3,  0,  0,
7120     21,  0,  0, 15,  0, 27,  0,  1,  1,  0,  0,
7130     21,  0,  0, 15,  0, 27,  0,  1,  1,  0,  0,
7140     21, 13,  0, 15,  0,  2,  0,  1,  0,  0,  0,
7150 /* CON FBL  SM PAN
7160     0,  4, 15,  3}
7170 m_fmvset(71,v)
7180 /*
7200 /*  AR D1R D2R  RR D1L  TL  RS MUL DT1 DT2 AME  BRASS
7210 v={ 20,  2,  0,  2,  1, 30,  0,  4,  2,  0,  0,
7220     21,  0,  0,  8,  0,  0,  0,  4,  2,  0,  0,
7230     20,  2,  0,  2,  1, 34,  0,  4,  3,  0,  0,
7240     21,  0,  0,  7,  0,  0,  0,  4,  3,  0,  0,
7250 /* CON FBL  SM PAN
7260     4,  7, 15,  3}
7270 m_fmvset(72,v)
7280 /*
7300 /*  AR D1R D2R  RR D1L  TL  RS MUL DT1 DT2 AME  CHORD 1
7310 v={ 24,  9,  3,  1,  2, 25,  0,  2,  0,  0,  0,
7320     31,  0,  0,  1,  0, 25,  0,  4,  0,  0,  0,
7330     23,  0,  0,  1,  0, 32,  0,  1,  0,  0,  0,
7340     27,  9,  0,  6,  2,  1,  0,  1,  0,  0,  0,
7350 /* CON FBL  SM PAN
7360     2,  6, 15,  3}
7370 m_fmvset(73,v)
7380 /*
7400 /*  AR D1R D2R  RR D1L  TL  RS MUL DT1 DT2 AME  PIANO
7410 v={ 21, 21,  5,  3,  3,  4,  0, 12,  7,  0,  0,
7420     31, 13,  8,  6,  0,  2,  0,  4,  3,  0,  0,
7430     31, 23,  5,  3,  3, 12,  0,  4,  3,  0,  0,
7440     31, 13,  8,  6,  0,  2,  0,  4,  7,  0,  0,
7450 /* CON FBL  SM PAN
7460     4,  4, 15,  3}
7470 m_fmvset(74,v)
7480 /*
7500 /*  AR D1R D2R  RR D1L  TL  RS MUL DT1 DT2 AME  SOLO dt.
7510 v={ 21, 13,  0,  0,  1, 20,  0,  3,  3,  0,  0,
7520     21,  0,  0,  0,  0, 28,  0,  1,  0,  0,  0,
7530     21,  0,  0,  0,  0, 27,  0,  1,  0,  0,  0,
7540     21, 13,  0,  7,  0,  2,  0,  1,  0,  0,  0,
7550 /* CON FBL  SM PAN
7560     0,  4, 15,  3}
7570 m_fmvset(75,v)
7580 /*
7600 /*  AR D1R D2R  RR D1L  TL  RS MUL DT1 DT2 AME  CHORD 2
7610 v={ 18, 16,  8,  0,  2, 27,  0,  3,  3,  0,  0,
7620     18,  0,  0,  0,  0, 26,  0,  1,  0,  0,  0,
7630     18,  0,  0,  0,  0, 27,  0,  1,  0,  0,  0,
7640     18,  9,  6,  8,  1,  0,  0,  1,  0,  0,  0,
7650 /* CON FBL  SM PAN
7660     0,  7, 15,  3}
7670 m_fmvset(76,v)
7680 /*
7700 /*  AR D1R D2R  RR D1L  TL  RS MUL DT1 DT2 AME  CHORD 3
7710 v={ 23,  1,  0,  2,  1, 25,  0,  2,  3,  0,  0,
7720     15,  1,  0,  6,  1,  6,  0,  2,  7,  0,  0,
7730     21,  1,  0,  2,  1, 25,  0,  1,  7,  0,  0,
7740     15,  1,  0,  6,  1,  6,  0,  1,  3,  0,  0,
7750 /* CON FBL  SM PAN
7760     4,  6, 15,  3}
7770 m_fmvset(77,v)
7780 /*
7800 /*  AR D1R D2R  RR D1L  TL  RS MUL DT1 DT2 AME  CHORD 4
7810 v={ 23,  1,  0,  2,  1, 25,  0,  8,  3,  0,  0,
7820     15,  1,  0,  6,  1,  8,  0,  8,  7,  0,  0,
7830     21,  1,  0,  2,  1, 25,  0,  4,  7,  0,  0,
7840     15,  1,  0,  6,  1,  8,  0,  4,  3,  0,  0,
7850 /* CON FBL  SM PAN
7860     4,  6, 15,  3}
7870 m_fmvset(78,v)
7880 /*
7900 /*  AR D1R D2R  RR D1L  TL  RS MUL DT1 DT2 AME  BELL
7910 v={ 31, 14,  3,  2,  2, 35,  0, 15,  3,  0,  0,
7920     31, 13,  6,  6,  4,  3,  0,  1,  7,  0,  0,
7930     31, 14,  3,  6,  3, 36,  0, 14,  7,  0,  0,
7940     31, 13,  6,  6,  4,  3,  0,  8,  3,  0,  0,
7950 /* CON FBL  SM PAN
7960     4,  4, 15,  3}
7970 m_fmvset(79,v)
7980 /*
8000 /*  AR D1R D2R  RR D1L  TL  RS MUL DT1 DT2 AME  BRASS
8010 v={ 17,  0,  0,  0,  0, 30,  0,  4,  3,  0,  0,
8020     17,  0,  0,  5,  0,  4,  0,  4,  3,  0,  0,
8030     17,  0,  0,  0,  0, 34,  0,  4,  1,  0,  0,
8040     17,  0,  0,  5,  0,  4,  0,  8,  1,  0,  0,
8050 /* CON FBL  SM PAN
8060     4,  7, 15,  3}
8070 m_fmvset(80,v)
8080 /*
9900 endfunc
10000 /*
10010 /* PLAY DATA
10020 /*
10100 func MUS()
10110 /*
10120 m_tempo(93)
10970 /*
10980 /*    OPM BASS
10990 /*
11000 p(0)="r8@70o2q8p3l16@k-6
11010 p(1)="r4[do]v15r*384
11020 p(2)="|:26g:||:<g>gg:|
11030 p(3)="|:25g:|<g>gggga-a
11040 p(4)="o2|:|:3b-<b->:|b-b-&|:3b-<b->:|b-b-:|
11050 p(5)="o3|:3c+<c+>:|c+c+&|:3c+<c+>:|c+c+
11060 p(6)="|:"+p(5)+":|
11070 p(7)=">g-8<e>g8<f>a-<d-> a8<g>b-8<a->b-<e>
11080 p(8)="|:e-8<e->e-8&<e->e-e- e-8<e->|1e-8<e->e-e-:|
11090 p(9)="|2e-&d-<d->d-<e->:|
11100 p(10)="|:4"+p(8)+"|2e-8<e-re->:|:|
11110 p(11)="|:|:a-a-a-r8a-8<a->a-r8a-<a->a-a-r a-a-a-|r4@b0,-33
00,0~3a-^4_3@bq2f<q8b->q2ffq8:|
11120 p(12)="r8.q2f<q7a->rq8a-8<q4a->q8b-8<q4b-q8>r:|
11130 p(13)="|:<|:16c:|>|:16b-:||:16a:|||:8a-:||:8b-:|:|
11140 p(14)="a-8.a-8a-8.|:4rb-:|
11300 p(30)="[loop]
11500 o={0,1,2,3,4,6,4,5,7,8,9,10,11,12,13,14,30,255}
11600 t(1)
11970 /*
11980 /*    MELODY 1
11990 /*
12000 p(0)="r8@71o3@v127q8p3l16@k-2@m9@h30@s5
12010 p(1)="r^4[do]r*372
12020 p(2)="(f,g)3&g*129d(g<c)4&c*8d>d<(d,g)4&g*188(g>g)3&g*141d
(g,a)3&a*9ga<d*180cd
12030 p(3)="|:e-8.dr8f8.>b-b-r<a-b-r(g>b-)3&b-*9
12040 p(4)="<(d,b-)5&b-*6r*13a-rga-rg8(a-g)
12050 p(5)="|f4e-ff+8.>br8(b<b)3&b*33b-r8a-b-r8
12060 p(6)="g-rfrg-a-c+(g<c)3&c+*81r>(c+>e-)3&e-*9<:|
12070 p(7)="f8.re-f g-fg-a-8g-a-8 b-a-b-b8b-b8
12080 p(8)="~1>g-(g-,b)3&b*9(b<e)3&e*9(e>g)3&g*9(g<c)3&c*9(c,f)3
&f*9(c>a-)3&a-*9(a-<d-)3&d-*9> a(a<d)3&d*9(d,g)3&g*9(g>b-)3&b-*9
(b-<e-)3&e-*9(e-,a-)3&a-*9(a->b)3&b*9(b<e)3&e*9b-*384
12090 p(9)="@72@v125o3|:>b-8.&<b-*120a8b-a8.f8.c8>b-ab-<crf8g
12100 p(10)="a-*156g8a-g8.|e-8.>b-*120<:|e-*108{e-&f&g&a-f&g&a-&
b-}4
12110 p(11)="~2<c2r>(b,b-)3&b-*9ra-_8g~8a-g_7r ~7a-a-b-a-ge-(e->
b-)2&b-*10(b-<c)3&c*9rc>b-8.ga-(a-<e-)3&e-*9_7r~7
12120 p(12)=">g&a-<&(d,f)3&f*9_7r~7>g&a-<&(e,g)3&g*21a-ge-e-f_7r
~7 g8a-ge-e->b-_7r~7<(g,b-)3&b-*21
12130 p(13)="(c>b-)2&b-*10(g-<a-)2&a-*10(c>b-)2&b-*10(f<g)2&g*10
(c>b-)2&b-*10(e-<f)2&f*10(c>b-)2&b-*10
12140 p(14)="<e-^4d8&e-f4.._7r~7 e-^4d8e-&(e-,b-)2&b-*34>b-^4
12150 p(15)="<e-&d&e-f8e-&fg8a-ge-8&(e->b-)2&b-*22(b-<b-)3&b-*15
3
12160 p(16)="@71o4c8d8v16
12170 p(17)="|:efecrc:|
12180 p(18)="(d>e)4&e*8fecrcefecrc
12190 p(19)="(c,g)2&g*22fe(e<c)3&c*93 (c>g)2&g*34fre8.c2(c,g)2&g
*34fre8.
12200 p(20)="<c2d2
12210 p(21)="<c8&(c>e)3&e*9(e,a-)3&a-*9r<c8.
12220 p(22)="|:3rd:|rd^64r32.
12500 o={0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,19,20,18,19
,21,22,30,255}
12600 t(2)
12970 /*
12980 /*    MELODY 2 (ECHO & SUB)
12990 /*
13000 p(0)="r4@75o3@v120q8p3l16@k7 @m9@h30@s3
13080 p(8)=">g-b<e>g<cf>a-<d->a<dg>b-<e-a->b<eb-*360r16
13090 p(9)="@72v11~2o3|:>b-8.&<b-*120a8b-a8.f8.c8>b-ab-<crf8g
13160 p(16)="@75o4c8d8v14r16
13220 p(22)="v12q5|:4dr:|v13q8r
13600 t(3)
13970 /*
```

```
13980 /*     BRASS & E.PIANO
13990 /*
14000 p(0)="r8q8116@k-4@m9@h30@s5
14010 p(1)="r4[do]
14020 p(2)="r*1152@79o5v14p2
14030 p(3)="18|:e-.d>b-16<f*312|g-.>b.<b*312:|g-.a-16^4b-.b16^1^
4 r*384116
14090 p(9)="@72@m8o2@v119|:g8.&<g*120f8gf8.c8.>a8 gfga r<d8e-
14100 p(10)="f*156e-8fe-8.>|b-8.g*120:|b-*108<{c&d&e-&fd&e-&f&g}
4 @72v11o3r*1p2~1
14160 p(16)="r*47@m@74o2v15|:c8e*168>b-8<e*168>a8<d*168|>a-8<e-*
72>g8<d*72:|
14170 p(17)=">v14a-2|:4rv13gv8:|
14500 o={0,1,2,3,9,10,11,12,13,14,15,16,17,30,255}
14600 t(4)
14970 /*
14980 /*     MELODY 3 & E.PIANO
14990 /*
15000 p(0)="r8q8116@k3 @m9@h30@s5
15010 p(1)="r4[do]@m8@71o3v11r*420p1
15020 p(2)="(f,g)3&g*129d(g<c)4&c*8d>d<(d,g)4&g*188(g>g)3&g*141d
(g,a)3&a*9ga<d*168@79o5v13p1
15030 p(3)="18|:e-.d.>b-16<f*300|g-.>b.<b*312:|g-.a-16^4b-.b16^1
^4 r*384116
15090 p(9)="r16@72@m8o2@v115|:g8.<g*120f8gf8.c8.>a8 gfga r<d8e-
15110 p(11)="r*1523@m
15120 p(12)="@74o2v15|:rd8(g-,g)4&g*164d8(g-,g)4&g*164c8(g-,g)4&
g*152|rc8(e,f)4&f*68>b-8<(e,f)4&f*56:|
15130 p(13)="v14f2|:4rv13dv8:|
15500 o={0,1,2,3,9,10,11,12,13,30,255}
15600 t(5)
15970 /*
15980 /*     FM CHORD 1
15990 /*
16000 p(0)="r8@76o4q8p3116@k-2@m17@h23@s3
16010 p(1)="r4[do]r*384v16
16020 p(2)="|:f8e_7e~7p1fp2ep3_7e~2e_7ep2~6e_7ep3_2e~15p1fp2ep3_
7e~7|c*204:|d*204v13@73
16030 p(3)="|:e-8e-8e-e-r|e-8e-fre-fr8:|f*108
16040 p(4)="|:g-8g-8g-g-r|g-8g-a-rg-a-r8:|a-*108
16050 p(5)="g-8g-8g-g-rg-8g-a-rg-a-r8 18e.f.f+g.g+.ae-*384
16060 p(6)="@80@m9v12o3116|:>b-8.<r *120a8r a8.r8.c8>r ar <c_5r~
5r8g r *156g8r g8.|r 8.>b-*120<:|r *108{e-rgr fra-r }4@m
16070 p(7)="@76o3q8@v123|:4ggfr8g8gfr8gggfrggfr4fr4|r4:|e-8f8
16080 p(8)="@73o4@v120|:p1ecrp2ercrcp1gp2erp1g8rdr drdp2r8>p1b-r
b-<p1gp2gr8p1g8dr
16090 p(9)="p1g8drp2ga8rp1grg8p2rg8p3a |p2f2f2:|@76o4v14c2|:4rv1
5fv8:|o4@76
16500 o={0,1,2,3,4,3,5,6,7,8,9,30,255}
16600 t(6)
16970 /*
16980 /*     FM CHORD 2
16990 /*
17000 p(0)="r8@76o4q8p3116@k8 @m17@h23@s3
17020 p(2)="|:c8c_7c~7p1cp2cp3_7c~2c_7cp2~6c_7cp3_2c~15p1cp2cp3_
7c~7|>a*204<:|>b*204<v13@73
17030 p(3)="|:c8c8ccr|c8cdrcdr8:|d*108
17040 p(4)="|:e-8e-8e-e-r|e-8e-fre-fr8:|f*108
17050 p(5)="e-8e-8e-e-re-8e-fre-fr8 18>b.<c.c+d.d+.e>b-*384
17060 p(6)="@80@m9v12o3116|:>r 8.<b-*120r8b-r8.f8.r8>b-rb-<r_5r~
5f8r a-*156r8a-r8.|e-8.>r*120<:|e-*108{r fra-rgr b-}4@m
17070 p(7)="@76o3q8@v123|:4b+b+b-r8b+8b+b-r8b+b+b+b-rb+b+b-r4b-r
4|r4:|a-8b-8
17080 p(8)="@73o4@v120|:p1grdp2gr8drp1ep2erp1e8erc g>b-<rp2e8p1r
dp1rdp2dr8p1d8re
17090 p(9)="p1c8rp2gc8.gp1cdr8p2ac8p3rp1c4..p3e-p1 |d2:|@76o4|:4
rv15dv9:|o4@76
17600 t(7)
17970 /*
17980 /*     FM CHORD 3
17990 /*
18000 p(0)="r8@76o3q8p3116@k3 @m17@h23@s3
18020 p(2)="|:g8g_7g~7p1gp2gp3_7g~2g_7gp2~6g_7gp3_2g~15p1gp2gp3_
7g~7|f*204:|g*204v13@73
18030 p(3)="|:a-8a-8a-a-r|a-8a-b-ra-b-r8:|b-*108
18040 p(4)="|:b8b8bbr|b8b<d->rb<d->r8:|<d-*108>
18050 p(5)="b8b8bbrb8b<d->rb<d->r8 18f+.g.g+a.a+.be-*384
18060 p(6)="@77p3o1v10116|:o3g*156f8gf8.c8.>a4..<d8e-f*156e-8fe-
8.>b-*156:|
18070 p(7)="@76o4q8@v123|:4e-e-dr8e-8e-dr8e-e-e-dre-e-dr4dr4|r4:
|c8d8
18080 p(8)="@73o4@v120|:p1cr8p2cr4p1cp2crp1c8r8.>  b-r8p2b-8r8.p
1b-p2b-r8p1b-8r8
18090 p(9)="p1a8r8p2ar8.p1ar8.p2ra8p3ra-4.<d8> |b-2:|@76o3|:4rv1
5b-v9:|o3@76
18600 t(8)
18970 /*
18980 /*     ADPCM RHYTHM
18990 /*
19000 p(0)="r8o0p3116@d1
19010 p(1)="r*336[do]rc d*2d*10r <e{fg} g>e*2e*10
19020 p(2)="|:4cce8.cer cre|crcer:|f{rccc}8ee
19030 p(3)="|:4crerccec rce8.ce|r:|a
19040 p(4)="|:crerccec rce8.cer:|
19050 p(5)="|:crecrc|ec:|{eeee}8
19060 p(6)="|:|:e-cc:|e-c:||:4crecr|rec:|ee<a>
19070 p(7)="|:4crec8.|ec:|ee
19080 p(8)="|:4crec|r8ec:|<e8>ee
19090 p(9)="|:4crec8.|ec:|{eece}8
19100 p(10)="|:cce-8.cec|creccrec:|crer8cer
19110 p(11)="|:cce-8.cec|:crec:|:|
19120 p(12)="|:cce-8.ceccre|ccrec:|r{ceee e<ef>r}4
19130 p(13)="|:4cce8.cec|rce8.cer:|ececrcee
19140 p(14)="|:3cce8.cecrce8.cer:|crecrcee|:4rc:|
19150 p(15)="|:12c-r:|
19500 o={0,1,2,3,4,5,6,7,8,7,9,10,11,10,12,13,14,15,30,255}
19600 t(9)
19970 /*
19980 /*     LA HIHAT
19990 /*
21000 p(0)="r8@127o4v8@u127q8p3116@d1
21010 p(1)="r8^1{dddd@v120d}*144r4.[do]
21020 p(2)="q7@v47@p48|:8d_15ddd~15|~11d_11d~6d_6d:|~14q7{dddd}8
_14drq7
21030 p(3)="|:12~14d_14ddd.~10d~17@128q5f8q8_17@127d_10:|r*384
21040 p(4)="q8@v51|:4|:@p88rdrd|:3@p48_5d~5@p88dr|d:||@128~17f_1
7@127:|d:|
21050 p(5)="@p88rdrd|:3@p48_5d~5@p88dr|d:|@128~17f_17@127
21060 p(6)="@p88rdrd@p48_5d~5@p88drd@p48@128~17f_17@p88@127drd@p
48_5d~5@p88drd
21070 p(7)="q5|:8r@128~17f8_17@127d@p48d@p88drdr@128~17f8f8_17@1
27drd:|
21080 p(8)="@v45@p50|:30~12d_12d~9d_9d:|
21090 p(9)="@128~58p3q7|:3rd:|rd8.r*264@v110@127d2
21500 o={0,1,2,3,4,5,6,7,8,9,30,255}
21600 t(11)
21970 /*
21980 /*     LA CRASH 1  (LEFT)
21990 /*
22000 p(0)="r8@126o4@u93q8p211@d1k3
22010 p(1)="r4[do]r*384v13
22020 p(2)="@p110fr2..@p90>~15b8_15<@p68gr@p110f*1152
22030 p(3)="@p110r8.{cc} ~8@p92~16c8.^8>a4.<_8g8_8@p110f*384_18
@p77g*1536~10
22040 p(4)="@p110|:4f*576|r:|@p70_6f8.~2@p110f16^2.
22500 o={0,1,2,3,4,30,255}
22600 t(12)
22970 /*
22980 /*     LA CRASH 2  (RIGHT)
22990 /*
23000 p(0)="r8@126o4@u93q8p111@d1k1
23010 p(1)="r4[do]r*384v13
23020 p(2)="|:@p30f*768:|r*384
23030 p(3)="@p30(aa) ~8g4.@p61e8^8.@p38c8.@p30g8c*384_18 @p56f*1
536~10
23040 p(4)="@p30|:f*768:|@p60_10|:f*576|r:|~6@p50e
23600 t(13)
23970 /*
23980 /*     LA CHORD 1
23990 /*
24000 p(0)="r8o4@u111@p74@k-7r*2
24010 p(1)="r4[do]r1@2r1v9116q8
24020 p(2)="|:f8e_17e~17fe _17e~12e_17e~12e_17e~12e ~15fe _17e~1
7|c*204:|~6d*204_6
24030 p(3)="|:|:e-8e-8e-e-r|e-8e-fre-fr8:|f*108
24040 p(4)="||:g-8g-8g-g-r|g-8g-a-rg-a-r8:|a-*108:|
24050 p(5)="g-8g-8g-g-rg-8g-a-rg-a-r8 18e.f.f+g.g+.aq8e-*384
24060 p(6)="@51@p73o5@v41116|:'b->b-'156'a8>a''b->b-''a8.>a''f8.
>f''c>c'84'f8>f''g>g''a->a-'156'g8>g''a->a-''g8.>g''e->e-'156:|
24070 p(7)="@22116o3q8@p40@v70|:4ggfr8g8gfr8gggfrggfr4fr4|r4:|e-
8f8p3
24080 p(8)="@2o4@v8211|:'cdeg',12'>b-<deg''>a<cdg'|'>a-2<ce-f''>
g2b-<df':|'>a-2<c',0r2@p60
24500 o={0,1,2,3,4,5,6,7,8,30,255}
24600 t(14)
24970 /*
24980 /*     LA CHORD 2
24990 /*
25000 p(0)="r8o4@u107@p54@k-7r*2
25020 p(2)="|:c8c_17c~17cc _17c~12c_17c~12c_17c~12c ~15cc _17c~1
7|>a*204<:|~6>b*204<_6
25030 p(3)="|:|:c8c8ccr|c8cdrcdr8:|d*108
25040 p(4)="||:e-8e-8e-e-r|e-8e-fre-fr8:|f*108:|
25050 p(5)="e-8e-8e-e-re-8e-fre-fr8 18>b.<c.c+d.d+.e>q8b-*384
25060 p(6)="@51@p36v5116~5|:o3g*156f8gf8.c8.>a4..<d8e-f*156e-8fe
-8.>b-*156:|
25070 p(7)="@22116o3q8@p66@v72|:4b+b+b-r8b+8b+b-r8b+b+b+b-rb+b+b
-r4<b->r4|r4:|a-8b-8
25080 p(8)="r*1536@p50
25600 t(15)
25970 /*
25980 /*     LA CHORD 3
25990 /*
26000 p(0)="r8o3@u107@p60@k-7r*2
26020 p(2)="|:g8g_17g~17gg _17g~12g_17g~12g_17g~12g ~15gg _17g~1
7|f*204:|~6g*204_6
26030 p(3)="|:|:a-8a-8a-a-r|a-8a-b-ra-b-r8:|b-*108
26040 p(4)="||:b8b8bbr|b8b<d->rb<d->r8:|<d-*108>:|
26050 p(5)="b8b8bbrb8b<d->rb<d->r8 18f+.g.g+a.a+.bq8e-*384
26060 p(6)="116@34@p50v11o5_5|:'>b-8.g''b-g'120'a8f''gb-''a8.f''
f8.c''c8>a'>'gb-''fa''gb-''a<c'r<'f8d''ge-'
26070 p(7)="'a-f'156'g8e-''a-f''g8.e-'|'e-8.>b-''>b-g'120:|'e->b
-'108r4
26080 p(8)="@22o4q8@p110@v72|:4e-e-dr8e-8e-dr8e-e-e-dre-e-dr4dr4
|r4:|c8d8
26090 p(9)="r*1536@p70
26500 o={0,1,2,3,4,5,6,7,8,9,30,255}
26600 t(16)
26970 /*
26980 /*     LA RHYTHM  (トラック9と似ている)
26990 /*
29000 p(0)="r8o2p3116@d1v13@u90
29010 p(1)="r*336[do]rc d*2d*10v16@u127cv14@u90r8.d*2d*10
29020 p(2)="|:4ccd8.cdr crdc|rcdr:|{rccc}8dd
29030 p(3)="|:4crdrccdc rcd8.cd|r:|r
29040 p(4)="|:crdrccdc rcd8.cdr:|
29050 p(5)="|:crdcrc|dc:|{dddd}8
29060 p(6)="|:|:dcc:|dc:||:4crdcr|rdc:|ddr
29070 p(7)="|:4crdcr8|dc:|dd
29080 p(8)="|:4crdc|r8dc:|r8dd
29090 p(9)="|:4crdcr8|dc:|{ddcd}8
29100 p(10)="|:ccd8.cdc|crdccrdc:|crdr8cdr
29110 p(11)="|:ccd8.cdc|:crdc:|:|
29120 p(12)="|:ccd8.cdccrd|ccrdc:|r{cddd drrr}4
29130 p(13)="|:4ccd8.cdc|rcd8.cdr:|dcdcrcdd
29140 p(14)="|:3ccd8.cdcrcd8.cdr:|crdcrcdd|:4rc:|
29150 p(15)="~12|:12cr:|_12
29500 o={0,1,2,3,4,5,6,7,8,7,9,10,11,10,12,13,14,15,30,255}
29600 t(19)
29970 /*
29980 /*     PCM E.ORGAN & E.PIANO 1
29990 /*
```

▶ZMUSIC.FNCとMUSICZ.FNCとは，苦し紛れのネーミングだな。
戸野　一博(19)北海道

```
30000 p(0)="r8@38o5v10@u90q3@p82l16@k2
30010 p(1)="r8gg[do]|:@p92g@p83g@p74<c>@p65g<@p56c@p47c>@p38g@p2
9g @p38g@p47g<@p56c>@p65g<@p74c@p83c>|@p92g@p101g:|@p92<c>@p83b
30020 p(2)="v11@q9|:|:@p95g@p85g<@p75c@p65e>@p55g<@p45c>@p35g@p2
5g
30030 p(3)="|1<@p35c>@p45g@p55g<@p65c@p75c>@p85g@p95g<@p105c>:|
30040 p(4)="|2@p35g<@p45c>@p55g<@p65c@p75c@p85c>@p95g<@p105c>:|:
|
30050 p(5)="@p80|:|:4b-fb-b-e-fga-:|
30060 p(6)="||:4<c+>g+<c+c+>|f+g+a+b:|f+bb-a-:|
30070 p(7)="|:<c+>g+<c+c+>f+g+a+b:|
30080 p(8)="r1@10q5o5v12_5@p60|:10r8.e-_25e-4~25e-_25e-8~25e-_25
e-4~25:|
30090 p(9)="l16o3q8@p90@v78|:4b+b+b-r8b+8b+b-r8b+b+b+b-rb+b+b-r4
<b->r4|r4:|a-8b-8
30100 p(10)="r*768@38o5@p82r*768q3v11
30500 o={0,1,2,3,4,5,6,7,8,9,10,30,255}
30600 t(20)
30970 /*
30980 /*    PCM E.ORGAM & E.PIANO 2
30990 /*
31000 p(0)="r8@39o6v10@u90q3@p44l16@k-6
31010 p(1)="r8.c@p57c_20@p61c~20g@p71g@p81c@p91f[do]@p81r@p71c@p
61d@p51c@p41g@p31g@p21c@p31f @p41r@p51c@p61d@p71c@p81g@p91g@p101
c@p91f @p81r@p71c@p61d@p51c@p41g@p31g@p21c@p31f @p41e@p51d
31020 p(2)="v11@q9|:|:@p41g@p51c@p61g@p71g@p81c@p91f@p101e@p91d
31030 p(3)="|1@p81e@p71c@p61d@p51e@p41f@p31e@p21d@p31g:|
31040 p(4)="|2@p81c@p71e@p61d@p51e@p41f@p31e@p21d@p31e:|:|
31050 p(5)="|:|:4|:5e-:|fe-f:|
31060 p(6)="||:4|:5f+:||g+f+g+:|bb-a-:|
31070 p(7)="|:|:5f+:|g+f+g+:|@35@p10q8v7r1
31080 p(8)="o3e-*384 o6@v30@p120|:b-*156a8b-a8.f8.c*84f8ga-*156g
8a-g8.|e-*156:|e-2.
31090 p(9)="@10l16o4q8@p25@v78r|:4e-e-dr8e-8e-dr8e-e-e-dre-e-dr4
dr4|r4:|c8d8
31100 p(10)="r*768@39o6@p44r*768q3v11r@p55c@p65g@p75g@p85c@p95f
31600 t(21)
31970 /*
31980 /*    PCM BASS
31990 /*
32000 p(0)="r8@23o1@u101q8@p68l16
32010 p(1)="r4[do]r*384v11
32020 p(2)="|:26g:||:<@p74v13@u127g>v11@p68@u101gg:|
32030 p(3)="|:25g:|<@p74v13@u127g>v11@p68@u101gggga-a
32040 p(4)="o1|:|:b-<@p74@u120b-@p68@u101>:|b-<@p74@u125b-@p68@u
101>b-b-&|:b-<@p74@u120b-@p68@u101>:|b-<@p74@u125b-@p68@u101>b-b
-:|
32050 p(5)="o2|:c+<@p74@u120c+@p68@u101>:|c+<@p74@u125c+@p68@u10
1>c+c+&|:c+<@p74@u120c+@p68@u101>:|c+<@p74@u125c+@p68@u101>c+c+v
12
32060 p(6)="|:"+p(5)+":|
32070 p(7)="@20>g-8<@u124e@u90>g8<@u124f@u90>a-<@u124d-@u90> a8<
@u124g@u90>b-8<@u124a-@u90>b-<@u124e@u90>@23
32080 p(8)="|:e-8<@p74@u127e-@p68@u101>e-8<@u90e->@u101e-e- e-8<
@p74@u127e-@p68@u101>|1e-8<@p74@u127e-@p68@u101>e-e-:|
32090 p(9)="|2e-&d-<@p74@u127d-@p68@u101>d-<@p74@u127e-@p68@u101
>:|
32100 p(10)="|:4"+p(8)+"|2e-8<@p74@u127e-@p68@u101>e-<@p74@u127e
-@p68@u101>:|:|@u101
32110 p(11)="|:|:v14a-a-a-r8a-8<v12@p74@u127a-@p68@u101v14>a-q1v
12@u110a-a-v14q8@u101a-<v12@p74@u127a-@p68@u101v14>a-a-r a-a-a-|
r4v11@u127@b0,-3300,0a-^4@b@p74@u127q1f<q7b->q1ff@p68q8@u101v12:
|
32120 p(12)="r8.v12@p74@u124q1fq8<a-@p68>q1a-@u101q8v14a-8q5<v12
@p74@u127a-@p68@u101v14>q8b-8q7<v12@p74@u127b-@p68@u101v14>q8r:|
@u101v12
32130 p(13)="|:<|:16c:|>|:16b-:||:16a:|||:8a-:||:8b-:|:|
32140 p(14)="a-8.a-8a-8.|:4rb-:|
32500 o={0,1,2,3,4,6,4,5,7,8,9,10,11,12,13,14,30,255}
32600 t(22)
32970 /*
32980 /*    PCM CHORD & STR 1
32990 /*
34000 p(0)="r8@10o4@u116l16@k0r*1
34010 p(1)="r2[do]r2.r1v7q7
34020 p(2)="@p29|:f8e_17e~17fe _17e~12e_17e~12e_17e~12e ~15fe _1
7e~17|c*204:|d*204
34030 p(3)="|:|:e-8e-8e-e-r|e-8e-fre-fr8:|f*108
34040 p(4)="||:g-8g-8g-g-r|g-8g-a-rg-a-r8:|a-*108:|
34050 p(5)="g-8g-8g-g-rg-8g-a-rg-a-r8 l8e.f.f+g.g+.ae-*380@35r*4
o5@p74@v25l16q8
34060 p(6)="|:b-*156a8b-a8.f8.c*84f8ga-*156g8a-g8.e-*156:|
34070 p(7)="r*1530@10r*6
34080 p(8)="o4@v69q6@p80|:ecrercrcgerg8rdr d>r<dr8>b-<r>b-<ggr8g
8dr
34090 p(9)="g8drga8rgrg8rg8gf2 |f2:|o4~6|:4rf:||:4_19f:|o4
34500 o={0,1,2,3,4,5,6,7,8,9,30,255}
34600 t(23)
34970 /*
34980 /*    PCM CHORD & STR 2
34990 /*
35000 p(0)="r8@10o4@u116l16@k7r*1
35020 p(2)="@p80|:c8c_17c~17cc _17c~12c_17c~12c_17c~12c ~15cc _1
7c~17|>a*204<:|>b*204<
35030 p(3)="|:|:c8c8ccr|c8cdrcdr8:|d*108
35040 p(4)="||:e-8e-8e-e-r|e-8e-fre-fr8:|f*108:|
35050 p(5)="e-8e-8e-e-re-8e-fre-fr8 l8>b.<c.c+d.d+.e>b-*380@37r*
4o5@p48@v47l16q8
35080 p(8)="o4@v69q6@p47|:grdgr8dreere8erc g>b-<re8>r<d>r<ddr8d8
re
35090 p(9)="c8rgc8.gcdr8ac8rc4..e- |d2:|o4~6|:4rd:||:4_19d:|o4
35600 t(24)
35970 /*
35980 /*    PCM CHORD & STR 3
35990 /*
36000 p(0)="r8@10o3@u116l16@k5r*1
36020 p(2)="@p68|:g8g_17g~17gg _17g~12g_17g~12g_17g~12g ~15gg _1
7g~17|f*204:|g*204
36030 p(3)="|:|:a-8a-8a-a-r|a-8a-b-ra-b-r8:|b-*108
36040 p(4)="||:b8b8bbr|b8b<d->rb<d->r8:|<d-*108>:|
36050 p(5)="b8b8bbrb8b<d->rb<d->r8 l8f+.g.g+a.a+.b@35@p116q8o3v7
e-*384
36060 p(6)="@p60v5l16|:o5g*156f8gf8.c8.>a4..<d8e-f*156e-8fe-8.>b
-*156:|
36080 p(8)="o4@v69q6@p63|:cr8cr4ccrc8r8.> b-r8b-8r8.b-b-r8b-8r8
36090 p(9)="a8r8ar8.ar4a8ra-4.<d8> |b-2:|o3~6|:4rb-:||:4_19b-:|o
3
36600 t(25)
36980 /*
36990 /*
40000 endfunc
40010 /*
50000 /* 注) 完全な演奏を聞くためにはCM64相当品が必要です。
50010 /*    コードのパートはLAとPCMを重ねているのでよく似た
50020 /* 部分がけっこうありますが、全く同じとは限らないので注意
50030 /* して下さい。
50040 /*    ZMUSICを使っているとはいえ、プログラムは巨大で
50050 /* 見にくいのでカウンターの値をよく見て入れて下さい。
```

## リスト2 DRAGON SABERコンフィグファイル

```
1    =solids_.pcm
.O0c =dsb.pcm
.O0c-=12,v130
.O0d =dsnar.pcm,v210,m1
.O0e =dsnar.pcm,v145,m1
.O0e-=16,mO0c
.O0f =16,v70,mO0c
.O1d =atkt4.pcm,p4,v140
.O1d-=26,v90
.O1e =atkt3.pcm,p2,v140,mO1d-
.O1e-=28,v90
.O1f =atkt2.pcm,p1,v140,mO1e-
.O1f+=29,v90
.O1g =atkt1.pcm,p-3,v140,mO1f+
.O1a =atkt2.pcm,p1,v150
.O0a =33,mO0e
.erase 1
.erase O1d-
.erase O1e-
.erase O1f+
```



## リスト3 すき

```
 10 /*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/
 20 /*                 す  き                */
 30 /*                                       */
 40 /*              By M.Watanabe            */
 50 /*                                       */
 60 /*    Programed by T.Sakai    H3/10/1    */
 70 /*                                       */
 80 /*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/
 90 m_init()
100 str p(99)[256]
110 str A[256],B[256],C[256],D[256]
120 str E[256],F[256],G[256],H[256]
130 str I[256],J[256],K[256],L[256]
140 str M[256],N[256]
150 char o(255),v(4,9),voi(4,10)
160 /*
170 for i=1 to 8
180  m_alloc(i,5500)
190  m_assign(i,i)
200 next
210  VD()
220  PD()
230 m_play()
240 end
250 /*
260 /*
270 func t(tt)
280  r=0
290   while o(r)<>255
300    m_trk(tt,p(o(r)))
310   r=r+1
320  endwhile
330 endfunc
340 /*
350 /*
360 func set(vn)
370  voi(0,0)=(v(4,1)*8)+v(4,0)
380  voi(0,1)=15
390  voi(0,9)=3
400   for x=0 to 3
410    for y=0 to 9
420     voi(x+1,y)=v(x,y)
430    next
440   next
450  m_vset(vn,voi)
460 endfunc
470 /*
480 /*
490 func VD()
500 /****** S n a r e   D r u m ************************
510 v={ 31,  0,  0,  1,  0,  0,  0, 15,  0,  3,
520     31, 17, 14,  8,  1,  2,  0,  2,  0,  0,
530     31, 20,  0,  9, 15, 10,  0,  2,  0,  0,
540     31, 14,  0,  8, 15,  2,  0,  3,  0,  0,  4,  7}
550 set(69)
560 /****** C r a s h ************************************
570 v={ 30,  4,  0,  2, 15, 13,  0,  1,  7,  1,
580     30,  2,  0,  1, 15, 17,  0,  4,  0,  2,
```

▶辛島さんが載ったということは，次にくるのは岡村孝子さんかな。それとも遊佐未森さんかな，う～ん楽しみ。レベッカもいいぞ，あと沢田聖子さんも……。

藤原　望満(17)岡山県

```
590      31,  8,  0,  4, 15, 27,  0,  9,  0,  2,
600      28, 12,  0,  5, 15,  0,  0,  1,  3,  0,  3,  7}
610 set(70)
620 /****** B a s s   D r u m ************************
630 v={ 31,  0,  0,  8,  0,  0,  0,  0,  0,  0,
640      31, 26,  6,  8, 10,  9,  0,  0,  0,  0,
650      31, 15,  0,  8, 15,  9,  0,  0,  0,  0,
660      31, 15,  0,  8, 15,  0,  0,  0,  0,  0,  4,  0}
670 set(71)
680 /****** T o m   T o m ****************************
690 v={ 19, 12,  9,  8, 12, 42,  1,  4,  3,  0,
700      15, 11, 12,  8, 12, 36,  1,  0,  3,  0,
710      15, 12,  9,  8, 10, 18,  1,  0,  3,  0,
720      15, 11,  9,  8, 10,  2,  1,  1,  3,  0,  4,  5}
730 set(72)
740 /****** H i - H a t   O p e n ********************
750 v={ 31, 31,  0, 15,  1,  7,  1, 11,  0,  3,
760      31, 26,  2, 15,  4, 16,  2, 12,  0,  3,
770      31, 22,  0, 15,  4, 10,  1,  1,  0,  1,
780      31, 31,  8, 15,  0,  4,  2,  7,  0,  0,  1,  5}
790 set(73)
800 /****** S t r i n g s  1 *************************
810 v={ 26,  5,  0,  2,  3, 28,  0,  1,  3,  0,
820      12, 31,  0,  6,  1,  6,  1,  1,  3,  0,
830      26,  5,  0,  2,  3, 21,  0,  1,  7,  0,
840      12, 31,  0,  6,  1,  6,  0,  4,  7,  0,  4,  7}
850 set(74)
860 /****** V o c a l ********************************
870 v={ 31, 24,  0, 10,  0, 40,  2,  7,  3,  0,
880      31, 24,  0, 10,  0,  7,  2,  1,  1,  0,
890      31, 31,  0, 10,  0, 25,  2,  1,  0,  0,
900      31, 31,  0, 10,  0,  7,  2,  2,  5,  0,  4,  0}
910 set(75)
920 /****** B e l l **********************************
930 v={ 14, 14,  0,  2, 15, 50,  1, 12,  3,  0,
940      14, 10,  0,  3, 15,  3,  1,  4,  0,  0,
950      10, 14,  0,  2, 15, 57,  1,  4,  0,  0,
960      16,  9,  0,  3, 15,  4,  2,  1,  0,  0,  4,  5}
970 set(76)
980 /****** S t r i n g s  2 *************************
990 v={ 27,  2,  1,  1,  2, 26,  1,  1,  3,  0,
1000     12,  4,  1,  9,  5,  8,  1,  1,  3,  0,
1010     27,  2,  1,  1,  2, 26,  1,  1,  7,  0,
1020     12,  4,  1,  9,  5,  8,  1,  1,  7,  0,  4,  7}
1030 set(77)
1040 /****** S t r i n g s  3 *************************
1050 v={ 20,  8,  0,  2,  1, 26,  0,  1,  0,  0,
1060     14,  5,  0,  7,  1, 20,  0,  2,  0,  0,
1070     12,  2,  0,  7,  1, 39,  1,  2,  0,  0,
1080     14,  5,  0,  7,  1, 12,  0,  2,  0,  0,  5,  7}
1090 set(78)
1100 /****** E . B a s s ******************************
1110 v={ 31, 12,  0,  6,  2, 46,  0, 10,  0,  0,
1120     31, 11,  4,  6,  2, 45,  0,  0,  7,  0,
1130     31,  8,  4,  6,  2, 20,  1,  0,  3,  0,
1140     31,  8,  3,  6,  2,  0,  1,  0,  0,  0,  0,  1}
1150 set(79)
1160 /****** G u i t a r ******************************
1170 v={ 31, 31,  0,  4,  5, 20,  2,  3,  3,  0,
1180     31, 31,  0,  4,  5,  0,  2,  1,  3,  0,
1190     31, 28,  2,  9,  5,  0,  2,  1,  0,  0,
1200     28, 14,  2,  9,  5,  1,  2,  3,  0,  0,  4,  0}
1210 set(80)
1220 /****** P i a n o ********************************
1230 v={ 31,  5,  2,  1,  1, 35,  1,  1,  7,  0,
1240     25,  5,  2,  1,  1, 45,  2, 13,  0,  0,
1250     31,  5,  2,  1,  1, 38,  1,  3,  3,  0,
1260     20,  7,  2,  6, 10,  5,  2,  1,  0,  0,  2,  6}
1270 set(81)
1280 /****** S y n . **********************************
1290 v={ 22, 14,  9,  6,  4, 31,  1,  5,  0,  0,
1300     28, 17, 10,  5,  3,  8,  1,  1,  0,  0,
1310     28, 17,  6,  6,  3, 43,  1, 10,  0,  0,
1320     28, 17, 10,  5,  3,  8,  1,  1,  0,  0,  4,  7}
1330 set(82)
1340 /****** D . G u i t a r **************************
1350 v={ 31,  6,  0, 15, 15, 17,  0,  1,  0,  0,
1360     31, 15,  0, 15,  0, 25,  0,  1,  0,  0,
1370     31, 15,  0, 15,  0, 21,  0,  1,  0,  0,
1380     31, 15,  0, 15,  0,  6,  0,  1,  0,  0,  0,  6}
1390 set(83)
1400 /*
1410 endfunc
1420 /*********************************************
1430 /*
1440 func PD()
1450 /*
1460 /*        D r u m s
1470 /*
1480 A="@71y2,3d4@69y2,9g4
1490 B="@71y2,6dd@69y2,9g@71d
1500 C="@71y2,6d4@69y2,9g.@71d16
1510 D="@71y2,6dd@69y2,9gy2,6r
1520 E="@71y2,6d4@69y2,9g4
1530 F="@71y2,6dd@69y2,9g4
1540 G="@71y2,3d4@69y2,9g.@71d16
1550 H="@71y2,6dd@69y2,9g@73y2,6o5co2
1560 I="@71y2,6d4@69y2,9g@73y2,6o5co2
1570 J="y2,6r16@71d16d@69y2,9g@71y2,7d
1580 /*
1590 L=G+B+"|:5"+E+F+":|"+C+B
1600 M="@71y2,6d4@69y2,5g4|:3"+I+E+":|"+I
1610 N=G+B+C+J+C+J
1620 /*
1630 p(0)="t127 @70 v15 y48,0 q8 p3 y3,3
1640 p(1)="l16o3r1r1r1 r4.a8&a2 @72o2|:r8y2,62a64y2,62a..y2,63f
64y2,63f..y2,64d64y2,64d..|1r2 r4.y2,63fy2,64dr2:|r4.y2,63fy2,64
dl8ry2,63f{y2,63ffy2,64dy2,64dy2,63ff}4|:y2,63f:|y2,64d4
1650 p(2)="l8"+A+B+C+D+E+B+C+"@72y2,6ry2,64fy2,9g@73y2,6o5co2
1660 p(3)=C+B+C+H+C+B+"@71y2,6d4@69y2,9g.@71y2,7d16y2,6r@69|:y2
,9g:|r
1670 p(4)=L+"@71y2,6d4@69y2,9g.@72y2,63f16y2,64dy2,64d16|:@69y2
,9g16:|r16@71d
1680 p(5)=L+"@71y2,6d4@69y2,9g4@72{y2,62ay2,63fy2,63fy2,63f}4y2
,64d4
1690 p(6)=G+"y2,6r@71d@69y2,9g@73y2,6o5co2"+E+F+C+"y2,6r@71d@69
y2,9g4"+E+B
1700 p(7)=M+"y2,6r4@69y2,9g4y2,9g16@72|:y2,63f16:|r16y2,64d@73y
2,6o5co2 y2,6r@69|:y2,9g:|@71dd@69y2,9gy2,5g4
1710 p(8)=G+B+C+"y2,6r16@71d16d@69y2,9g@73y2,6o5co2"+C+B+E+B
1720 p(9)=G+B+"|:"+C+"y2,6r16@71d16d@69y2,9g4:|
1730 p(10)="@71y2,6d4@69y2,9g|:3y2,6r24:|@71y2,6d@69y2,9g|:y2,9
g16:|r16@71d16
1740 p(11)=G+B+C+H+C+B+C+"r@71y2,64d@72y2,9f4
1750 p(12)=G+B+C+D+C+B+"@71y2,6d4@69y2,9g.@71y2,7d16y2,6r@72|:y
2,9f:|@71d
1760 p(13)=L+"@71y2,6d4@69y2,9g4@72y2,63fy2,63f16@69|:y2,9g16:|
r16@71d
1770 p(14)=M+"@71y2,6d4@69y2,9g@72ay2,9f|:y2,9d:||:y2,64d16:| y
2,9f@71|:y2,9d:|ddy2,9dy2,5d4
1780 p(15)="@71d@72y2,62a@69y2,9g@72|:y2,63f16:||:3y2,64d16:|r1
6y2,64d4
1790 p(16)="@71dl1o6@78v11d&d&d&f+2.l8o2|:3y2,7r24:||:y2,7r:|r4
|:15y2,7r:|r2.|:3y2,7r:||:16y2,7r32:|
1800 p(17)="|:y2,3@71d4r4:|@71y2,3d4y2,6r4@72|:y2,9f6:|y2,9d6@7
0y2,3o3a1o2
1810 p(18)="r2.y2,3r@69y2,9g@71d16@72|:y2,62a16:|r16y2,63fy2,64
d@72d@69y2,5gy2,9g4
1820 p(19)=N+"@71y2,6d4@69y2,9g.@71d16y2,6d@72|:y2,63f:|y2,64d
1830 p(20)=N+"@69|:y2,9g16:||:5y2,9g:||:y2,9g16:|r16@71d16
1840 p(21)=N+"@71y2,6d4@69y2,9g4@72|:y2,63f:||:y2,64d:|
1850 p(22)=N+"@71y2,6d4@69|:y2,9g:||:y2,9g16:|@72|:4y2,63f16:|y
2,64d
1860 p(23)="l8o2@71y2,3v13d4@69y2,9g.@71d16 @71v12y2,6dd@69y2,9
g@71d v10|:@71y2,6d4@69y2,16g.@71d16 y2,6r16@71d16d@69y2,16g@71y
2,7dv7:|v6 @71d4@69y2,16g4@72v11aaff @70o3v10a4@69o1v8g@71v6{rd}
 @69v714rgv6rgv4rgv2rgv1rg
1870 o={0,1,2,3,4, 5,6,7, 8,9,10,11,12, 13,6,14, 8,9,15,16,
1880     17,18,8,9,10, 19,20,21,22,21,22, 23,255}
1890 t(1)
1900 /*
1910 /*        B a s s
1920 /*
1930 A="l8o2brr.<b16>a+a+ra+ g+rg+4f+f+r.<f+16> err.<e16d+d+rd+
 c+rc+4>f+f+r4
1940 B="l8o3drr.d16>ee4ef+rr.{br}b16bb4f+rr.{bb}q2b16q8bb4ere4a
ra4
1950 /*
1960 p(0)="v15 y49,04 q8 p3
1970 p(1)="l2o2@74|:g+f+e.f+4g+f+er8f+4.:|
1980 p(2)="l8o2@79|:g+rr.<g+16>f+f+rf+ |1 err.<e16>ef+r4:|
1990 p(3)="err.e16ef+r4 g+rr.<g+16>f+f+rf+ err.<e16>ef+r4 g+rr.
<g+16>f+f+rf+ err.<e16>r@v127f+16r16f+4 v15
2000 p(4)="brr.<b16>a+a+ra+ g+rg+4f+f+r4 err.<e16d+d+rd+ c+rc+4
>f+f+r4"+A
2010 p(5)=A+A+"l8o3d+4r.d+16rd+&d+.<d+16>> g+4r.g+16g+a+<cd+
2020 p(6)="c+4r.c+16rc+&c+.<c+16>> f+4r.f+16f+g+a+&b ere.<e16d+
rd+4
2030 p(7)="c+rc+c+32r32<c+16>d+rd+4> ere.<e16d+rd+4 c+rc+4d+rd+
4> f+rf+rf+rf+r @v127ree4raa4 v15
2040 p(8)="l8o3|:drr.d16>ee4ef+rr.{br}b16bb4f+rr.{bb}q2b16q8bb4
ere4ara4:|
2050 p(9)="l8o2brr.<b16>aaragrr.<g16>gaa4brr.<b16>aaragrr.g.aa4
 g+rr.<g+16>f+f+rf+err.<e16>ef+r4g+rr.<g+16>f+f+rf+err.<e16>r@v1
27f+f+4v15
2060 p(10)="l2o2@74bag{rg}4a4bag{rg}4a4|:l8o2q6rb<f+d>ra<f+c+r>
g<d>b|1r2:|r4r16.@79{g&g+&a&a+&b&<c&c+&d}16&d16.
2070 p(11)="l2o3gf+e{ddd}2c+1&c+1&c+>@v127{ra}4a4v15
2080 p(12)="|:6"+B+":|
2090 p(13)="l8o3  drr.d16>v14ee4ev13f+rr.v12{br}b16bb4v10f+rr.v
9{bb}q2b16q8bb4v8ere4v7ara4
2100 p(14)="l8o3v6drr.d16>v5 ee4ev4 f+rr.v3 {br}b16bb4v2 f+rr.v
1{bb}q2b16q8bb4@v67ere4ara4
2110 o={0,1,2,3,4,5,6,7, 8,9,  5,6,7, 8,10,11,
2120     8, 12,13,14, 255}
2130 t(2)
2140 /*
2150 /*        V o c a l   a n d   e t c .
2160 /*
2170 A="|:d+d+d+d+:|     d+c+>bg+b4r{b<c+}|:5d+:|c+>bf+g+4<c+4c
+4>r{b<c+}
2180 B="|:d+d+d+{d+d+}:| d+c+>bg+b4r{b<c+}|:5d+:|c+>bf+g+4<c+4c
+4>r{b<c+}
2190 C="l2o5b8f+8f+.<dc+>b8<f+8f+r4>ba+g+8d+8d+r4ba+g+8<d+8d+2
2200 D="l8o4a4<d4edeb    a4f+ed4r4a4f+ed4>a4b<ee4r2>a4<d4edeba2
<d4.>ba4.a<d4.>bagf+gd4ed
2210 E="l8o4a4<d4edeb    a4f+ed4r4a4f+ed4>a4b<ee4r2>a4<d4edeba2
<d4.>ba4.a<d4.>bagf+gd4e
2220 F="l8o5v14d&d2..rv11a4f+ed4r4a4f+ed4>a4b<ee4r2>a4<d4edeba2
<d4.>ba4.a<d4.>bagf+gd4e
2230 /*
2240 p(0)="@77 v15 y50,20 q8 p3 l2 o5
2250 p(1)="r1r1r1r2.{d+e}8{f+g+a+}8 ba+g+8d+8d+.ba+g+8<d+8d+2.
2260 p(2)="@76l2o5|:ba+g+8d+8d+r4|1ba+g+8<d+8d+r4>:|ba+g+8<d+8d
+r8
2270 p(3)="@75l8o4{b<c+}"+A+"|:9d+:|ed+c+>b4&f+{g+f+}g+.<c+.>g+
f+.b.b<ed+c+>b<c+4r>{b<c+}"+A
2280 p(4)="d+d+d+{d+d+}|:5d+:|ed+c+>b4&f+.f+16g+.<c+.>g+f+f+b<c
+ed+c+>b<c+4r4
2290 p(5)="f+4..r4..f+f+ff+g+f+eed+e4..r4..eed+ef+ed+c+{c+c+}>b
{bb}r {f+g+}b{bb}r{f+g+}bb<c+d+c+>bb|:{f+g+}b{bb}r:|{f+g+}bb<c+d
+c+>bf+<f+&f+1
2300 p(6)="@80l8o5q6rg+g+4rbq8b4@75@v127"+D
2310 p(7)="ed2..@76v15"+C+"r8@75l8o4{b<c+}
2320 p(8)=B+p(4)
2330 p(9)="f+4..r4..f+f+ff+g+f+e4d+e4..r4..{ee}8ed+e{f+f+}ed+c+
{c+c+}>b{bb}r {f+g+}b{bb}rf+bb<c+d+c+>bb|:{f+g+}b{bb}r:|f+bb<c+d
+c+>bf+<f+&f+1
2340 p(10)="ed2..&d2r2r1r1o3q6@82|:rb<df+>ra<c+f+>rgb<f+r2:|
2350 p(11)="@78l8o5q8a1&a2{agf+ef+g}2v14b1&b1r2r@74o4@v127gg4@7
5
2360 p(12)="|:4"+D+":|
2370 p(13)="l8o4v14a4<d4v13edeb v12a4f+ev11d4r4v10a4f+ev9d4>a4v
8b<ee4r2>v6a4<d4v5edebv4a2<v3d4.>bv2a4.a<v1d4.>@v67bagf+g
```

▶「MUSIC PRO-68K［MIDI］」のバージョンアップは出ないんですか。僕はMMLよりも譜面入力派なので，トリルとかベンドなんかの記号があって，それを置くだけでそのとおり演奏してくれたりするとすごくうれしいのにな。　本田　登(19)長野県

```
 2380 o={0,1,2,3,4,5,6,7, 8,9,6,10,11. 12,13,255}
 2390 t(3)
 2400 /*
 2410 /*        V o c a l   e c h o   a n d   e t c .
 2420 /*
 2430 p(0)="@77 v10 y51,32 q8 p3 l2 o5 r8
 2440 p(6)="@80l8o5q6 ee4rgq8g4r@75"+D
 2450 p(7)="ed2..@76v9 "+C+"r8@75v10l8o4{b<c+}
 2460 p(10)="ed2..&d2r4.r1r1 l8o3q6@82v15|:rf+b<d>rf+a<c+>rdg<dr
2:|
 2470 p(11)="@78v11q8l8o5ra1&a4.v15>{agf+ef+g}2v14e1&e1r2r@74o4@
v127dd4@75v10r8
 2480 p(12)=E+"|:3"+F+":|
 2490 p(13)="l8o5v13d&d2..rv9a4f+ev7d4r4a4f+ev6d4>a4v5b<ee4r2>v4
a4<d4v3edebv2a2<d4.>v1ba4.a<v0d4.>bag
 2500 o={0,1,2,3,4,5,6,7, 8,9,6,10,11, 12,13,255}
 2510 t(4)
 2520 /*
 2530 /*        C h o r d   1
 2540 /*
 2550 p(0)="@81 v15 y52,08 q8 p3
 2560 p(1)="l2o4|:ba+g+8<v10d+8d+2>v15a+4 ba+g+8<d+8d+4.>a+4.:|v
14
 2570 p(2)="l2o4|:3ba+g+{ra+}4a+4:|ba+g+{rg+}4g+4
 2580 p(3)="l2o4|:f+f+f+f+g+f+g+{bb}:|@74v13
 2590 p(4)="v14l8o4a+4r<v15{f+f+}>v14ra+4.f+1g+4r.<c+16>rg+4.e1
 2600 p(5)="v15|:g+rg+rf+rf+rererf+rf+r:|erererer
 2610 p(6)="l8o4v13q8f+@v127q7bb4r<dq8d4
 2620 p(7)="@74v14l2o3ab<ef+aag<{dd}>>ab<ef+aa{gf+de}1
 2630 p(8)="l2o4f+f+f+{rf+}4f+4f+f+f+r8f+4. d+d+d+{rf+}4f+4d+d+d
+{re}4e4 v13
 2640 p(9)="@81v15l8o5d2c+2>bf+f+4rb<c+4d2c+2>a<f+f+4r>b<c+4
 2650 p(10)="f+>b<df+f+>a<c+f+c+>bb4r<c+>b<c+f+>b<df+f+>a<c+f+db
b2r4
 2660 p(11)="@74v13l2o4gf+g<{ddd}2>v14g1&g1rr8@74v15o3b8b4
 2670 p(12)="@74v13l2o3av12b<v11ev10f+v9av8av7g<v6{dd}>>v4ab<v2e
f+v1aa@v67{gf+de}1
 2680 o={0,1,2,3,3,4,5,6,7,8, 3,4,5,6,7, 9,10,11,
 2690     7, 7,7,7, 12,255}
 2700 t(5)
 2710 /*
 2720 /*        C h o r d   2
 2730 /*
 2740 p(0)="@81 v15 y53,28 q8 p3
 2750 p(1)="l2o4|:g+f+{d+d+}4d+f+4g+f+{d+d+}4d+4r8f+4.:|v14
 2760 p(2)="l2o4|:3g+f+d+{rf+}4f+4:|g+f+d+{re}4e4
 2770 p(3)="l2o4|:d+c+d+d+d+d+>b<{g+g+}:|@74v13
 2780 p(4)="v14l8o4f+4r<v15{c+c+}>v14rf+4.d+1e4r.g+16re4.c+1
 2790 p(5)="v15|:ererd+rd+rc+rc+rd+rd+r:|>brbrbrbr
 2800 p(6)="l8o4r@v127q7g+g+4rbq8b4
 2810 p(7)="@74v14l2o4r1c+def+d{bb}r1c+def+>{bbbb}1
 2820 p(8)="l2o4dc+>b<{rc+}4c+4dc+>b<r8c+4.> ba+g+<{rd+}4d+4>ba+
g+{rb}4b4 v13
 2830 p(9)="@81v15l8o4b2a2{f+r}4r2a4b2a2{f+r}4r2a4<dr4.c+r4.r4>f
+4rar4<dr4.c+r4.>br2..
 2840 p(10)="@74v13l2o4ddd{aaa}2v14c+1&c+1r.@78v15{b<c+def+g}4
 2850 p(11)="@74v11l2o4r1c+v10dv9ef+v8dv7{bb}r1v4c+v3dv2ev1f+>@v
67{bbbb}1
 2860 o={0,1,2,3,3,4,5,6,7,8, 3,4,5,6,7, 9,10,
 2870     7, 7,7,7, 11,255}
 2880 t(6)
 2890 /*
 2900 /*        C h o r d   3
 2910 /*
 2920 p(0)="@81 v14 y54,08 q8 p3
 2930 p(1)="l2o4|:d+c+>{br}4<rc+4d+c+>{br}4<r4.c+4.:|v13
 2940 p(2)="l2o4|:3d+c+>b<{rc+}4c+4:|d+c+>b{rb}4b4
 2950 p(3)="l2o4|:ba+bbbbg+<{ee}>:|@74v12
 2960 p(4)="v13l8o4c+4rv14{a+a+}v13rc+4.c1>b4<r.c+16r>b4.a+1
 2970 p(5)="@81v15l8o4|:9brbr:|br@77o5{ed+c+>ba+g}4
 2980 p(6)="l8o4q6f+@74q7ee4rg q8@78{b<c+def+g}4
 2990 p(7)="@78v15q8l4o5a1&a2.<d>a1gf+d{def+g}l2a.<d4ef+ef+{gf+d
e}1
 3000 p(8)="@74v14l2o3bag{ra}4a4bagr8a4. g+f+e{ra+}4a+4g+f+e{rg+
}4g+4 v12
 3010 p(9)="@81v14l2o4f+f+{dr}4rf+4f+f+d8rr8f+4l8br4.ar4.r4d4rf+
r4br4.ar4.f+r4.
 3020 p(10)="@83v12l8o5d4&v13d&v14d&|:4y54,232c+16&y54,08d16&:|d
1
 3030 p(11)="v14l8o4{agf+ef+g}2>g1&g2.&v13|:4y54,232f+16&y54,08g
16&:|v12g4.
 3040 p(12)="@80v15l8o5ee@78v11o4{b<c+d}8
 3050 p(13)="@78v14q8l4o5a1&v12a2.<v11d>v10a1v8gf+dv7{def+g}l2v5
a.<v4d4ev3f+v2ev1f+@v67{gf+de}1
 3060 o={0,1,2,3,3,4,5,6,7,8, 3,4,5,6,7,9,10,11,12,
 3070     7, 7,7,7, 13,255}
 3080 t(7)
 3090 /*
 3100 /*        E t c .
 3110 /*
 3120 p(0)="@74 v11 y55,16 q8 p3
 3130 p(1)="l2o2|:g+f+e.f+4g+f+er8f+4.:|
 3140 p(2)="l1o5rrrr2.@80v13f+8&{f+&f&e&d+&d}8
 3150 p(3)="l1o5rrrr2r8  f+8f+8&{f+&f&e&d+&d}8
 3160 p(4)="l8o4|:7r1:|r4.{c+&d+}8q6f+f+q8f+16&g+.
 3170 p(5)="l8o5|:7r1:|r2..q1{eee}q8a+&{a+&a&g+&g&f+}r2.
 3180 p(6)="r2..q1{eee}q8g+&{g+&g&f+&f&e}r2.r1@76v15l4o5|:g+rf+r
brf+r:|
 3190 p(7)="l8o5e2..@77v9{ed+c+>ba+g+}4f+r2r@78{b<c+def+g}4
 3200 p(8)="@78v9q8l4o5a1&a2.<d>a1gf+d{def+g}l2a.<d4ef+ef+{gf+}d
4e8
 3210 p(9)="@76v15l8o6dv11d4.v15c+v11c+4.r1r1r2.
 3220 p(10)="@80v13o5a&{a&g+&g&f+&f}8r1r1r1r2rf+f+&{f+&f&e&d+&d}
8
 3230 p(11)="@78v15q8l1o6d&d&d&f+2.r4r1r1r1r2
 3240 p(12)="@83v8l8o5rd4&v9d&v10d&|:4y55,240c+16&y55,16d16&:|d2
..
 3250 p(13)="v14l8o3{agf+ef+g}2c+1&c+2.&v13|:4y55,240c16&y55,16c
+16&:|v11c+2
 3260 p(14)="@78v10o4r8{b<c+def+g}4
 3270 p(15)="@78v9q8|:4l4o5a1&a2.<d>a1gf+d{def+g}l2a.<d4ef+ef+{g
f+de}1:|
 3280 p(16)="@78v8q8l4o5a1&a2.<v6d>v5a1v4gv3f+v2dv1{def+g}l2@v69
a.<d4@v67ef+ef+
 3290 o={0,1,2,3,4,5,6,7,8,9,10, 5,6,7,8,11,12,13,14,
 3300     15, 16,255}
 3310 t(8)
 3320 endfunc
```

## リスト4 THE ENTERTAINER

```
10 '┌─────────────────────────────┐
20 '| "The Entertainer" by Scotto Joplin |
30 '└─────────────────────────────┘
40 CLS0:TEMPO0:DEFINT a-z:DEFSTR p:DIM p(18)
50 MEM$(&HB190,36)=HEXCHR$("F200770C3201252F25009F9F5F9F04090407
03030503101030A5000000000000080000000")
60 PLAY STRING$(8,"i01:"):GOTO90
70 LABEL "!":FOR j=0 TO i:PLAY p(j);:NEXT:PLAY ":";:RETURN
80 '
90 n$="11111111":PRINT CHR$(26)
100 INPUT" 1.[4/4]   2.[24/16]  ";a:IF a<>2 THEN a=1
110 IF a=1 THEN PLAY "t97";
120 FOR k=1 TO 8:ON VAL(MID$(n$,k,1)) GOSUB HEX$(k):NEXT
130 IF a=2 THEN PLAY "[t70r16t140r16]";
140 PLAY"":END
150 '----
160 LABEL "1"
170 p(0)="k5 v11q8l16o6
180 p(1)="dec<a8bg8dec<a8bg8dec<a8baa-g4>>¯12q5g8q7<dd+
190 p=  "e>c8<e>c8<e>c&c4&c >cdd+ecde8<b>d8c4.<<dd+
200 p=p+"e>c8<e>c8<e>c&c4. agf+a>ce8dc<a>d4.<<dd+
210 p=p+"e>c8<e>c8<e>c&c4&c >cdd+ecde8<b>d8c4.cd
220 p=p+"ecde8cdcecde8cdc ecde8<b>d8
230 p(2)=p+"c4.<< dd+
240 p(3)=p+"c4&c<< eff+
250 p(4)="g8ag8eff+g8ag8>ec<g ab>cdedcd<g>efgagef
260 p(5)="g8ag8eff+g8ag8gaa+bb8b8af+dg4&g<eff+
270 p(6)="g8ag8eff+g8ag8>ec<g ab>cdedcdc4&c<gf+g
280 p(7)=">c8<a>c8<a>c<ag>ceg8ec<g a8>c8ed8c&c4.< dd+
290 p(8)=p+"c4.r8<<
300 p(9)="f8ef8ef8 ra>d<a>cdc<a g8f+g8f+g8 r>cecdedc
310 p(10)="d8c+d8c+d8 rfafgagf >ccc4<a8g8<ggg8g8
320 p(11)=p(9)
330 p(12)="<ag+a>g8f8c ed+ea8>c<ge c8c8ed8c&c4. <dd+
340 p(13)=p(8)
350 i=13:"!":RETURN
360 '
370 LABEL "2"
380 p(0)="k8 v8 q8l16o6r32.
390 "!":RETURN
400 '
410 LABEL "3"
420 p(0)="k5 v11q8l16o5
430 p(1)="dec<a8bg8dec<a8bg8dec<a8baa-g4>>>¯9q5d8q7rr
440 p=  "r2. reff+gefg8df8e4.r8
450 p=p+"r2r4. crcrrf+8rf+rf4.r8
460 p=p+"r2. reff+gefg8df8e4.r8
470 p=p+"r1  gefg8df8
480 p(2)=p+"e4.r8
490 p(3)=p+"e4&e<cdd+
500 p(4)="e8re8cdd+e8re8rrrr1>
510 p(5)="e8re8cdd+e8re8rrrgg8f+8rcr<b4&bcdd+
520 p(6)="e8re8cdd+e8re8rrrr1
530 p(7)="r1f+8f+8ff8e&e4.rr>
540 p(8)=p+"e4.r8<
550 p(9)="d8c+d8c+d8r8f4. e8d+e8d+e8r8g4.
560 p(10)="b8a+b8a+b8r8b4. >ccc4c8c8<eee8e8
570 p(11)=p(9)
580 p(12)="rrra8a8rr2r8f+8bb8g&g4.rr>
590 p(13)=p(8)
600 i=13:"!":RETURN
610 '
620 LABEL "4"
630 p(0)="k8 v8 q8l16o5r32.
640 "!":RETURN
650 '
660 LABEL "5"
670 p(0)="k5 v15q5l8o4r1r2.bq6rv14
680 p="o3c>c<g>c<f>c<e>c< <g>>c<<g>bc>cc<b
690 p=p+"c>c<g>c<f>c<ee-d>c<d>c<bgab
700 p=p+"c>c<g>c<f>c<e>c< <g>>c<<g>bc>cer
710 p=p+"ce<b->e<a>f<a->f<g>e<<g>b>c<gc
720 p(1)=p+"b
730 p(2)=p+"r
740 p(3)="c>c<<g>>c< c>c<<g>>c< <f>>f<f>f< e>e<<g>>e<
750 p(4)="c>e<<g>>e< c>e<ee- d>d<d>d d<fed
760 p(5)="c>c<<g>>c< c>c<<g>>c< <f>>f<f>f< e>e<c>e
770 p(6)="ffd+d+eeee c<abb>c<gcb
780 p(7)=p(2)
790 p(8)="<f>a<a>a <f>a<a>a c>c<<g>>c< c>c<<g>>c< <g>b<b>b<g>bdb
 >cc4ccr4.<
800 p(9)="<f>a<a>a <f>a<a>a c>c<<g>>c< c>c<<g>>c< fdefg>ed+e f+<
dgb>c<gcb
810 p(10)=p(2)
820 i=10:"!":RETURN
830 '
840 LABEL "6"
850 p(0)="k5 v15q5l8o4r1r2.gq6rv14
860 p="o3rg<g>b-<f>a<e>g rgrgrggg
870 p=p+"rg<g>b-<f>a<ee-d>arag<gab>
880 p=p+"rg<g>b-<f>a<e>g rgrgrg>c<r
890 p=p+"c>c<<b->>c<<a>>c<<a->>c<<g>>c<<g>gc<gc
900 p(1)=p+">g
910 p(2)=p+"r
920 p(3)="c>grg rgrg >rcrc rcrc
```

▶ (善) さんへ。「大江戸繁盛記」のエンディングはグラフィックが一枚出るだけです。
新保　敦工(16)新潟県

```
930 p(4)="rcrc rc<<ee- d>br>c<b<fed
940 p(5)="c>grgrgrg >rcrcrcrc
950 p(6)="cccccccc< ddggc<gc>g
960 p(7)=p(2)
970 p(8)=">rfrfrfrf rgrgrgrg rgrgrgrg f+f+4f+gr4.
980 p(9)=" rfrfrfrf rgrgrgrg <fdefg>>ccc d<<dgb>c<gc>g
990 p(10)=p(2)
1000 i=10:"!":RETURN
1010 '
1020 LABEL "7"
1030 p(0)="k5 v14q518o2r1r2.gq6rv14
1040 p="o3 rergrfre rerfreer
1050 p=p+" rergrfrr rf+rf+rrrr
1060 p=p+" rergrfre rerfregr
1070 p(1)=p+"rgrgrara rgrrrrrr
1080 p(2)=p(1)
1090 p(3)="rrrr rrrr rara- rgrg
1100 p(4)="rgrg rgrr rgra grrr
1110 p(5)="r1 rara-rgrb-
1120 p(6)="aaaagggg r1
1130 p(7)=p(1)
1140 p(8)="r1 rerererere rfrfrfrf d+d+4d+er4.
1150 p(9)="r1 rerererere r2rgf+garrrr2
1160 p(10)=p(1)
1170 i=10:"!":RETURN
1180 '
1190 LABEL "8"
1200 p(0)="k5 v14q518o1r1r2.gq6rv14"+STRING$(16,"r1")
1210 p(1)="r1r1r1o5d16dcr.r2
1220 p(2)="r1r1 <<fff+f+r2r1
1230 p(3)=STRING$(15,"r1")+">rrf16fe16&e4.r11rrrrrrrr
1240 i=3:"!":RETURN
1250 '
```

## （善）のゲームミュージックでバビンチョ

# さよなら1991

メリークリスマス＆あけましておめでとう。みんな風邪ひいて鼻水たらしてないか？　編集部はいま風邪の巣窟だぞ。くれぐれも身体には気をつけてほしい。

さて今月は，一応1991年が幕を閉じたわけということで，1991年に発売されたゲームミュージックCDを振り返ってみようと思う。

しかし，こんな私の独断と偏見だけのコーナーがよく続いてるなぁ。このコーナーが始まったのは1991年の3月号。あまり多くの人に読まれていないせいか，このコーナーについての反応はほとんどなく（連載当初は多かったが），カラーから白黒ページに移転されたときもほとんど反響がなかった（ような気がする）。「対談～」コーナーも，そのせいか1回きりで沈黙。ま，周りの無関心に便乗していられるからこそ続いているという説もあるけど。

で，以下はこれまでの本コーナーにおいて「お勧め度」が8以上だったタイトルを中心に挙げている。

買おうか買うまいか悩み続けて年が明けてしまったチミ，単にチェックし忘れているチミ，寒さで心までかじかんでしまっているチミ，TAKE A LOOKだぜよ。

**●アルシスベストセレクション　CD:PSCX-1016**
**ポリスター　2,400円(税込)**

タイトルとは裏腹に実は「プリンス・オブ・ペルシャ（PC-9801版）」が中心のこのアルバム。ほかにも「ナイトアームズ（X68000版）」と「スタークルーザー（メガドライブ版）」が収録されているけど，全曲じゃないのが残念（あ，だからベストセレクションなのか，納得）。

ナイトアームズとスタークルーザーは，ぜひ一度作曲者本人によるシンセアレンジを聴いてみたいなぁ。しみじみ……。

**●アクトレイザー/古代祐三　CD:ALCA-105**
**アルファレコード　2,800円(税込)**

**●交響組曲アクトレイザー/古代祐三**
**CD:ALCA-182　アルファレコード**
**2,500円(税込)**

前者は同名のスーパーファミコン用ゲームのオリジナルサウンドアルバム，後者はそのオーケストラアレンジバージョン集。どちらも内容的には同じだが，私は音的にも編曲にも，さらに磨きのかかった後者のほうをお勧めしたい。

それにしてもコシロンは，メロディとバックのからみ（いわゆるフィル）の構成が綺麗だねえ。

**●ナムコサウンドエクスプレスVOL.4「ドラゴンセイバー」　CD:VICL-40014～15**
**ビクター音楽産業　2枚組　3,000円(税込)**

2枚組のアルバム。1枚は，ドラゴンセイバーのために新たに書き下ろされたBGMが収録されていて，もう1枚は前作「ドラゴンスピリット」のシステムアレンジが収録されている。つまり，あの懐かしの名曲がFM＋PCMで蘇ってしまうわけなのだ。

ところで，今月のOh!X LIVEではZ-MUSIC用のデータとしてドラセイ4面の曲が掲載されているのでそちらもどうぞ。

それにしても去年いちばん聴いたCDかもなぁ。

**●ミスティ・ブルー/古代祐三　CD:ALCA-123**
**アルファレコード　2,000円(税込)**

アドベンチャーゲームのBGMということもあってか情景描写的な曲が中心。また，いくつかユーロビートっぽい曲もあって，それらはメロディアスでいつものコシロ節に近い。

おまけにアクトレイザーのサウンドボードアレンジが収録されている。

**●FORMULA/SEGA SST　CD:PCCB-00059**
**ポニーキャニオン　2枚組　3,200円(税込)**

セガの「ラッドモビール」「GPライダー」「R360」などの体感ゲームサウンドを中心としたアルバム。DISC1にアレンジバージョン，DISC2にオリジナルサウンドが収録されている。

アレンジバージョンは，原曲から近からず遠からずの絶妙なアレンジでとても気持ちがよい。アレンジバージョンがよすぎて原曲のほうが聴きづらい。はたしてこれはいいことなのか，悪いことなのか……？

**●STRIKE FIGHTER／SEGA SST**
**CD:PCCB-00067**
**ポニーキャニオン　1,500円(税込)**

オリジナルサウンドは，よくも悪くもいつものセガセガしい調子。やっぱりお目当てはアレンジバージョン。「BORN OUT」のアレンジバージョンのオルガンソロが最高。オリジナルサウンドのほうもこのくらいがんばってればなあ。

**● THUNDER ZONE / DATA EAST GAMADELIC　CD:PCCB-00068**
**ポニーキャニオン　1,500円(税込)**

オリジナルサウンドで，ディストーションギターがメロディを奏でているのには驚き。曲自体もヘビメタ，ブラックありと飽きさせない構成になっている。

ゲームミュージックがここまで進歩したかと実感させてくれる1枚だ。

**●出たな!!　ツインビー　CD:KICA-750**
**キングレコード　2,800円(税込)**

いかにもゲームミュージックっぽいメロディを展開しつつも，映画音楽のような壮大な雰囲気のフレーズやフィルを巧みに挿入したりと，かなり音楽的に優れたものといえそう。

コナミのゲームミュージックというと，やたらオケヒットがうるさいというイメージを持つ人もいるだろう（A-JAX,野獣の～）。が，そんなイメージはこの1枚がきっと吹き飛ばしてくれるだろう。

## そういうわけで

ERIC PERSINGというミュージシャンのアルバムって出てないかな。マイケル・ジャクソンなんかと共演したこともあるらしいんだけど。ローランドのD10やD5，U220，MT32などのデモ曲なんかも手掛けている人なんだよね（スタジオミュージシャン臭いんだよな，どうも）。誰か知ってたら教えてね。それでは，今年も（いつ打ち切りになるかわからない）このコーナーをよろしく。ハガキ待っているぞ。

# 吾輩はX68000である
## ［第9回］

# 完成！簡易アニメーションツール

Izumi Daisuke　泉　大介

遅ればせながらクリスマスプレゼントである
御仁の奮闘の結果，
楽しんでいただければ幸いである

吾輩のIOCSコールに用意されたグラフィック描画機能とグラフィック画面のスクロール機能をお届けしたところで，ちょっとしたサンプルをお目に掛けることができるようになった。うちの御仁がなんと，アセンブラでプログラムを作ってくれたのである。これまで一貫してアセンブラに背を向け続けてきた御仁に，いったいどんな心境の変化があったのかを吾輩が知る由もないが，どうやら事はOh!Dyna編集部でIBM PCのゲームに触れたことに端を発しているようである。

## 御仁のカルチャーショック

IBM PCには７万本のソフトがある（そうである）。売れ筋のLotus 1-2-3や，DTPソフトのAldus Pagemaker，フライトシミュレータやインディなど，ビジネスソフトやゲームソフトを合わせた総数なのだろうが，それにしても半端な数ではない。もちろん玉石混交である。そのソフトの山の中には極めてイロモノに近いソフトも存在する。本誌の副編集長を務め，現在Oh!Dynaの編集長であるN氏がこのイロモノに興味を持った。氏が集めたこれらのゲームは編集室のJ-3100で披露され，廊下を徘徊するあまたのライターたちの注目を集めるところとなる。そしてこの中にうちの御仁もいたのである。

画面ではミッキーマウスが昼寝をして夢を見ている。羊が柵を飛び越える例のあれである。そして，

ONE
TWO
THREE
⋮

なんとJ-3100が羊の数を数えているのである。ルーカスフィルムからサウンドボードが発売されており，これで音声を出力することができるのだ。出現する羊の数は数字キーを押して決めることができる。声に合わせて羊は柵を飛び越え，そして，押したキー番目の羊は柵に引っかかってしまうようになっている。ドテッとこけた羊に驚いて目を覚ましたミッキーは，「誕生パーティーを開こう」とばかりに街に買い物に出かけることになる。

道を歩いているときに数字キーを押すと，ミッキーは押した数字個の車輪を持つ乗り物に乗ることができる。「４」を押すと，

ONE, TWO, THREE, FOUR wheels.

という声とともに車輪が登場したあと，ミッキーは自動車に乗る。「１」なら一輪車，「３」ならサイドカーである。御仁はここではまってしまった。「５」はなんだろう。「７」は？　そして「９」は……。人を食った解答，そして延々と繰り返される数の読み上げ。これは小児向けの数の数え方ソフトなのである。このあとプレゼントを選び，招待状を出し，パーティーでは買ってきたものをみんなに分ける。いずれのシーンでも，ひたすら数を数え続けるだけ。取りえといえば，徹頭徹尾ストーリーがディズニーキャラクターのアニメーションによって進んでいくことくらいだろうか。パソコンがディズニー映画を上演しており，人がキーボードで（ただし０～９のキーしか受け付けないが）そのストーリーに介入できる，といった趣である。ただ数を数えるためだけにアニメーションし，J-3100にしゃべらせているのだ。

「こんなソフトが成立するのか……」

絶句した御仁の受けたカルチャーショックを想像されたい。日本という閉鎖された世界にいて，日本というアプリケーションどまりのなかで今日まで生きてきたのである。吾輩は最初からこのサウンドボード相当のものを内蔵している。にもかかわらず，こういった発想のソフトはない。さすがはセサミストリートを生み出したアメリカ産といったところか。ソフトの名前がまたすましている。1-2-3というのである。もちろん「こんなソフトが日本で売れるだろうか」という危惧は容易に想像できる（ゲーム小僧は見向きもしないだろう）が，御仁の知り合いの所帯持ちのライターのうちでは，子供が元気に1-2-3しているという話は添えておくべきだろう。かの子供はきっと「もじもじくん」でも元気に遊んでくれるのではないだろうか。吾輩も子供に殺戮を奨励するゲームばかりではなく，こんなソフトも欲しいところである。日

本にも「ポンキッキ」や「ピンポンパン」はあるではないか。

こういったN氏が披露したソフトの中に，今回お届けするプログラムの原形になったものがある。パラパラ漫画作成ソフトとでもいうべきもので，数枚の絵を用意しておき，それを順に表示することで簡単なアニメーションを実現するのだ。ひとつ前に描いた絵は薄い色で表示され，続きを次のページに書き込む手助けとなっている。「なんだ。これなら簡単じゃないか。『吾輩』のサンプルにももってこいだ」。御仁がそう思ったのかどうかは定かではない。ただ，それから一心にアセンブラでプログラムを作成してくれたことは確かである。そして，御仁が初めてアセンブラを使って作ってくれたプログラムが成功裏に終わったのも事実である。おかげをもって，今回諸兄にささやかなクリスマスプレゼントをお届けできる運びとなったわけだ。

## まずは習作

C言語には通じていても，そしてCコンパイラが吐き出したアセンブリ言語のプログラムを眺めることはあっても，はたまたZ80のアセンブラで連載を行っていたという実績はあっても，MC68000のアセンブラでプログラムを作成するのは初めての御仁であるから，まず最初に取り掛かったのは絵を描くための習作プログラムである。もちろん絵を描くのに吾輩が携えているマウスを使わない手はない。IOCSコールにはマウス関係の処理が用意されているので，まずは小手調べといったところか。

リスト1がそのプログラムである。御仁はアセンブラで作成したのだが，本稿はデバッガを使ってお届けしているのでデバッガ用に直してある。$100016_H$まではこれまでお届けしたグラフィック用のサンプルどおりである。ファンクションキーを消去し，画面モードを設定し，グラフィックをONにしている。

$10001E_H$で使っているIOCSコール$7D_H$は，ソフトウェアキーボードを表示しないようにしている。ご存じのようにマウスの右ボタンを押すと吾輩はソフトウェアキーボードを表示する。これを使って，キーボードが壊れても，あるいはキーボードとマウスは遠いと感じている向きにも，マウスだけで文字の入力ができるようになっているのである。ここではマウスの右ボタンを利用するため，このソフトウェアキーボードの処理をキャンセルするよう指示を出している。ちなみにパラメータは，

D1.l ＝ 0：消去
　　　　1：表示
　　　　2：表示されているか調べる
　　　－1：マウス右ボタンで制御
D2.l ＝ XXXX_YYYY$_H$
　　　　D1.l=1のときに座標を指定

となっている。D1.lに2を指定したときは，ソフトウェアキーボードが表示されているか否かが，D0レジスタに0（表示されていない），1（表示されている）で返される。

実をいうと，御仁は最初この部分を作成していなかった。御仁はMicro EMACSを使っているが，Micro EMACSがこの処理を行っているためチャイルドプロセスで実行している御仁はまったく問題に気づかなかったのである。プログラムが完成に近づいたあるとき，たまたまMicroEMACSを抜けて実行して初めて不備を発見したのであった。

これで右ボタン使用のマウスを使う用意は整った。さて，マウスの使用法であるが，これは，

1)　マウス初期化(IOCSコール$70_H$)
2)　マウスカーソル表示(IOCSコール$71_H$)

の2段階で行う。この2つのIOCSコールを順に実行するだけで，マウスカーソルが画面に表示され，以後マウスを使った処理ができるようになる。プログラムでいえば$100022$〜$100028_H$の間がこの処理にあたる。

マウスを使って絵を描くために，御仁はIOCSコール$B6_H$を使うことにしたようだ。すでにご紹介しているように，これは画面に点を表示するIOCSコールである。点を表示する座標と色をメモリに収め，そのアドレスをA1レジスタにセットして利用するようになっている。このプログラムでは$10002A_H$で，早々にA1レジスタにパラメータのアドレス$100070_H$をセットしておき，以後A1レジスタのデータを変更しないように作っている。御仁の基本方針は以下のようである。

1)　マウスの右ボタンが押されていれば終了
2)　マウスの左ボタンが押されていれば
3)　マウスカーソルの座標を取り出し
4)　IOCSコール$B6_H$で点を描く

これを延々と繰り返すわけである。

マウスのボタンが押されているかどうかをチェックするには，IOCSコール$74_H$を利用する。実行するとD0レジスタに，XX_YY_LL_RR$_H$の4バイトのフォーマットでマウスのデータが返されるIOCSコールである。XX，YYはX，Y方向の移動量，LL，RRは左右のボタンの状態で，押されていなければ$00_H$に，押されていれば$FF_H$になる。たとえばマウスカーソルがX方向に1ドット，Y方向に2ドット移動し，右ボタンが押されているなら，

D0.l＝010200FF$_H$

となるわけである。

$100034_H$ではBTSTという命令を使ってボタンが押されたかどうかをチェックしている。BTSTはデータを2進数で考えたときに特定の桁が0か1かを判定する命令である。マウスの右ボタンが押されていれば，D0.lには??????FF$_H$がセットされる。$FF_H$を2進数で書けば$11111111_B$となるので，この場合D0.lの0〜7桁目のいずれかを調べればボタンが押されたかどうかをチェックで

きる。調べた桁が0ならばゼロフラグがセットされる。したがって，

```
btst   #0,d0  ← 0桁目をチェック
bne    終了処理アドレス
```

とすれば，右ボタンが押されたら終了するというプログラムが可能となる。

100030～100040$_H$がボタン判定を行うループである。御仁は0桁目と8桁目をチェックすることで，右ボタンと左ボタンの判定を行うことにしたようだ。さて左ボタンが押されていれば，上の3)の手順に取りかかることになる。次の100044$_H$である。

マウスカーソルがいま表示されている座標は，IOCSコール75$_H$で知ることができる。D0.lにXXXX_YYYYというフォーマットで座標が返されるのでこれを利用すればいい。もしマウスカーソルが(16,256)にあるなら，D0.lに返されるデータは00100100$_H$となる(0010$_H$=16,0100$_H$=256)。ところで，点を描画するIOCSコールB6$_H$のパラメータは100070$_H$に格納してある。データは，

```
100070  dc.w  0  ← X座標
100072  dc.w  0  ← Y座標
100074  dc.w  15 ← 点の色
```

となっているから，IOCSコール75$_H$でD0.lに取り出されたマウスカーソルの座標は，

```
move.l  d0,$100070
```

とするだけで簡単にセットすることができる。パラメータのアドレスはA1レジスタにセットしてあるので，

```
move.l  d0,(a1)
```

でOKだ。あとはIOCSコールB6$_H$を実行して点を描くだけである。そして再びマウスのボタンチェックへとループすればいい。以上，御仁のリストの10004E$_H$まで終了である。

リストで残っているのは100052$_H$以降である。ここは右ボタンが押されたときの終了処理である。マウスの右ボタンが離されるのを待ち，マウスカーソルを消去。そしてファンクションキーを表示し直せば処理はすべて終了となる。ソフトウェアキーボードの処理を元に戻していないが，気になる方は自分で追加していただきたい。

さして長くもないプログラムなので，どうぞ入力して試してみていただきたい。この習作プログラムの問題点が明らかになるはずである。マウスで描画するプログラムをX-BASICなどで作成した経験のある諸兄は，実行せずともおわかりかもしれない。問題点はこのプログラムが「IOCSコール75$_H$を実行したときのマウスカーソルの座標」に点を描画するところにある。このことは，IOCSコール75$_H$が実行されたときマウスカーソルが前回描画した点の隣にいれば，そしてそのときだけ，諸兄は線を描けるということを意味している。実は左ボタンを押したままマウスをかなりゆっくりと動かさなければ，これは不可能なのである。たいていの場合，諸兄の描いた絵は点描になってしまうことだろう。御仁もしばらくは，「1ドットだけのエアブラシなんてめったにないぞ」とうそぶきながら使っていたのだが，だんだん非を認める気持ちが支配してきたらしい。基本は押さえたということで，作り直す気になったようだ。

## 目指すものは

吾輩は1024×1024ドットのグラフィック実画面をもっている。表示画面を256×256ドットとすれば，横方向に4画面分，縦方向に4画面分の，合計16画面をこの実画面に収めることができるわけである。前回グラフィック画面のスクロールを行うIOCSコールB3$_H$を取り上げたが，これを使えば16枚の絵を自由に表示することができるようになる。クリスマスツリーを背景に，

も・う・す・ぐ・X'mas

と順次表示していく絵だって可能だ。御仁の目指すプログラムはこれである。

プログラムの不備が明らかになったところで御仁のとったアプローチは，線を描画するIOCSコールB8$_H$を使うことである。IOCSコール75$_H$で得たマウスカーソルの座標が必ずしも連続した点であるとは限らないなら，線で結んでしまえばいい，という発想である。これは，前回描画した線の最後の座標を覚えておき，そこから新しく

**リスト1　画面に絵を描くための習作**

```
-an 100000
00100000    move.w  #3.-(sp)      ← fncキー非表示
00100004    move.w  #14.-(sp)
00100008    _concrtl
0010000A    addq.l  #4.sp
0010000C    move.w  #16.d1        ← 768×512ドット×16色
00100010    moveq   #$10.d0       ← 画面モード設定
00100012    trap    #15
00100014    moveq   #$90.d0       ← グラフィックON
00100016    trap    #15
00100018    move.l  #0.d1
0010001E    moveq   #$7d.d0       ← ソフトキーボード消去
00100020    trap    #15
00100022    moveq   #$70.d0       ← マウス初期化
00100024    trap    #15
00100026    moveq   #$71.d0       ← マウスカーソルON
00100028    trap    #15
0010002A    movea.l #$100070.a1   ← pset用データアドレス
00100030    moveq   #$74.d0       ← ボタンデータを得る
00100032    trap    #15
00100034    btst    #0.d0         ← 右ボタンが
00100038    bne     $100052       ← 押されたら終了
0010003C    btst    #8.d0         ← 左ボタンが
00100040    beq     $100030       ← 押されていないければループ
00100044    moveq   #$75.d0       ← 座標データを得る
00100046    trap    #15
00100048    move.l  d0.(a1)       ← xy座標セット
0010004A    moveq   #$b6.d0       ← 点を描画
0010004C    trap    #15
0010004E    bra     $100030       ← これを繰り返す
00100052    moveq   #$74.d0       ← ボタンデータを得る
00100054    trap    #15
00100056    btst    #0.d0         ← 右ボタンのチェック
0010005A    bne     $100052       ← 離されるまでループ
0010005E    moveq   #$72.d0       ← マウスカーソルOFF
00100060    trap    #15
00100062    move.w  #9.-(sp)      ← fncキー表示
00100066    move.w  #14.-(sp)
0010006A    _concrtl
0010006C    addq.l  #4.sp
0010006E    _exit
00100070    dc.w    0             ← X座標
00100072    dc.w    0             ← Y座標
00100074    dc.w    15            ← 色
00100076
```

得た座標へ線を引けば実現できる。ついでに，リスト1では右ボタンで終了となっていたのを，右ボタンで次の画面に移れるように変更すれば完成である。オマケ機能として，前の画面に描いた絵が新しい画面にもコピーされていると申し分ない。続きを描きやすくなるだろう。

これらの点を押さえながら御仁が作成したのがリスト2のプログラムである。例によってデバッガ用のプログラムに直してあるが，これまで紹介してきたリストとは少々異なる形式にしてある。リスト2はこれまで吾輩の機能紹介に使ってきたプログラムと比較してかなり大きなプログラムである。ここで，

bsr　　$1000dc

などとされた日には，諸兄が右に並んだアドレスの中から1000DC$_H$を探し出すのは大変な作業だろう（bsrについては後述する）。そこで，アセンブラで使われる「ラベル」を併用したリストにしてみたというわけである。

アセンブラを使用する場合には，どのアドレスにBSRするのか，あるいはBccするのかを，プログラムを作成している段階で決定することはできない。このため，アドレスの代わりに「newpage:」などというラベルを使用すると以前説明したことがあるのを覚えていらっしゃるだろうか。このリスト2では，BSRやBccを行う行は，

bsr　　$1000dc

というデバッガの表記と，

bsr　　newpage

というラベルを使った表記を併記してある。ラベルはアドレスとMC68000の命令の間の隙間にちょうどくるように入れてあるので，この隙間をざっと見渡して「newpage:」を探せばいい。かなり便利に探せるはずである。

**リスト2　画面に絵を描く**

```
-an 100000
        *
        *   メインルーチン
        *
00100000        move.w  #3,-(sp)        * fncキー非表示
00100004        move.w  #14,-(sp)
00100008        _conctrl
0010000A        addq.l  #4,sp
0010000C        moveq   #$af,d0         * _os_curof
0010000E        trap    #15
00100010        move.w  #2,d1           * 256×256ドット×16色×1
00100014        moveq   #$10,d0         * _crtmod
00100016        trap    #15
00100018        moveq   #$90,d0         * _g_clr_on
0010001A        trap    #15
0010001C        move.w  #0,d1           * 表示ウィンドウ左上x座標
00100020        move.w  d1,d2           *                  y座標
00100022        move.w  #1023,d3        *                右下x座標
00100026        move.w  d3,d4           *                  y座標
00100028        moveq   #$b4,d0         * _window
0010002A        trap    #15
0010002C        move.l  #0,d1           * ソフトキーボード消去
00100032        moveq   #$7d,d0         * _skey_mod
00100034        trap    #15
00100036        moveq   #$70,d0         * _ms_init
00100038        trap    #15
0010003A        moveq   #$71,d0         * _ms_curon
0010003C        trap    #15
0010003E        move.l  #0,d1           * (0,0)
00100044        move.l  #$ff00ff,d2     * (255,255)
0010004A        moveq   #$77,d0         * _ms_limit
0010004C        trap    #15
0010004E        movea.l #$100182,a1     * line用データアドレス
              † movea.l #lndata,a1
00100054        moveq   #0,d7           * d7はシーンナンバ
00100056        bsr     $1000dc         * ナンバに応じたページを表示
              † bsr     newpage
        loop:
```

さてリストであるが，複数の画面に絵を描けるようにする下準備として，グラフィックの描画範囲を拡大する必要がある。画面を256×256ドットモードにした場合，普通にやったのでは(0,0)－(255,255)の範囲にしか描画できない。そこで，10001C～10002A$_H$で，IOCSコールB4$_H$を使い(0,0)－(1023,1023)の範囲に描画できるよう設定し直している。パラメータはリストを参照されたい。また10003E～10004C$_H$では，マウスカーソルの移動できる範囲にも制限を加えている。

このリストには，複数の画面に絵を描けるようにするために，画面を切り替えるプログラムが用意されている。newpage(1000DC$_H$)がそれである。プログラムには，「何度も繰り返し現れる同じ処理」といったものが存在する。これをその都度入力するのは面倒このうえない。そこでサブルーチン呼び出しといわれる方法がMC68000に用意されている。これは「BSR」という命令で，Bccと同じように指定されたアドレスに分岐するが，処理が終わったあと戻ってくるという便利な命令である。1000DC$_H$から始まっているnewpageサブルーチンをちょっと見ていただきたい。1000FA$_H$でこのサブルーチンは終了するが，最後の命令が「RTS」となっているのを確認できるだろう。これがBSRと対をなす「呼ばれたところへ戻る」命令である。いずれこのあたりの仕組みは説明する予定だが，いまは「RTSはBSRで呼び出されたところに戻ってくる命令」とだけ覚えておいていただきたい。

newpageはD7.lにセットされているデータに従って表示画面を切り替えるサブルーチンである。D7.lに0がセットされていれば(0,0)－(255,255)の画面が表示され，D7.lに1がセットされていれば(256,0)－(511,255)の画面が表示されるように作られている。このサブルーチンを呼び出しているのが100056$_H$である。newpageが何を行っているのかわかってしまえば，

bsr　　newpage

と出てきても「あ，表示画面を切り替えるのか」と読むことができる。newpageサブルーチンがそのままここにはめ込まれているより，遥かにプログラムは読みやすくなるだろう。そのため「一度しか利用しないプログラムでも読みやすさを考えてサブルーチンにする」という手法がよく利用される。吾輩にはプログラムの読みやすさなど関係ないが，人間にとっては意味のあるアプローチといえよう。

10005A～100078$_H$がボタン判定である。先のリスト1ではBTSTを使ってボタンの判定をしていた御仁だが，ここではTSTに改めている。TSTはデータを調べる命令で，データの正負，0か否かを知ることができる。おそらく，データレジスタのデータを調べるBTST命令が4バ

イトの命令なのに対して，TSTは2バイトなのでプログラムが短くなると踏んだのであろう。また，BTSTより2.5倍高速に判定できるという点も考慮したに違いない。まぁ，こういう判定方法もあるということだ。吾輩はBTSTのほうが直接的でわかりやすくていいと思うのだが，御仁の趣味なのかもしれない。データは2の補数表現でチェックされるので，右ボタンが押されていれば(D0.b=$FF_H$なら)負の数として判定されるわけである。

ボタン判定は，

1) 右ボタンが押されたらloop1で処理
2) 左ボタンが押されたらdrawで処理
3) さもなければ(A1).lに−1をセット

の繰り返しで行われている。おそらく3)は諸兄を混乱させることだろう。A1にはラインを描くIOCSコール$B8_H$用のデータが入っているアドレスがセットされている。(A1).lはその先頭データで，ラインの始点XY座標がそれぞれワードデータで入っている。これをMOVE.lを使って一気に−1にしているのである。

御仁の戦略を説明しよう。御仁は，左ボタンを押しながらマウスを動かすとそれに合わせて線を引くようにこのプログラムを作っている。線を引くには始点と終点の2つの座標が必要である。始点の座標は前回の線の表示に用いた終点座標をそのまま使い，現在のマウスカーソル座標を改めて終点とすればこの2つの座標が得られる。ではマウスの左ボタンが離されたらどうするか。この場合は，次にマウスの左ボタンが押される座標を新しい始点として線を描き始めなければならない。さもなければ一筆書きしかできなくなってしまう。始点座標の−1は，マウスのボタンが離されたという印なのである。

これはプログラムのdrawの部分を見ていただければよりはっきりとする。ここでは，

1) マウスカーソルの座標を終点にコピー
2) 始点が負なら終点の座標を始点にコピー
3) 線を描画
4) 終点の座標を始点にコピーして次回の描画に備える

という順に処理が行われている。「move.l #−1,(a1)」の意味がおわかりいただけただろうか。

では続けて，このプログラムのもうひとつの山場であるendの部分を見ていただこう。ここは，

1) D7を1つ大きくし
2) 16になったら終了
3) そうでなければ現在表示されている絵をcopyサブルーチンで次の画面に移し
4) newpageサブルーチンで表示画面を次の画面に切り替える

という処理が行われている。

リストの順番でいくとまずはnewpageサブルーチンだが，これはIOCSコール$B3_H$を使い，D7.lに応じてグラフィック画面をスクロールさせる処理を行っている。ス

```
0010005A        moveq   #$74.d0         * _ms_getdt
0010005C        trap    #15
0010005E        tst.b   d0              * 右ボタンが
00100060        bmi     $100074         * 押された
              † bmi     loop1
00100064        tst.w   d0              * 左ボタンが
00100066        bmi     $10007c         * 押された
              † bmi     draw
0010006A        move.l  #-1.(a1)
00100070        bra     $10005a
              † bra     loop
        loop1:
00100074        bsr     $1000a0
              † bsr     end
00100078        bra     $10005a
              † bra     loop
        draw:
0010007C        moveq   #$75.d0         * _ms_curgt
0010007E        trap    #15
00100080        add.l   $10017e.d0
              † add.l   xystart.d0
00100086        move.l  d0.4(a1)        * xy座標セット
0010008A        move.w  (a1).d0         * 始点x座標が
0010008C        bpl     $100094         * 正の数だったら描画
              † bpl     draw1
00100090        move.l  4(a1).(a1)      * さもなければ終点座標をコピー
        draw1:
00100094        moveq   #$b8.d0         * _line
00100096        trap    #15
00100098        move.l  4(a1).(a1)
0010009C        bra     $10005a
              † bra     loop
        *
        *   終了処理
        *
        end:
001000A0        moveq   #$74.d0         * _ms_getdt
001000A2        trap    #15
001000A4        tst.b   d0              * 右ボタンが
001000A6        bne     $1000a0         * 押されたままならendへ
              † bne     end
001000AA        addq.l  #1.d7           * 次のシーンへ
001000AC        cmpi.w  #16.d7          * シーン15まで終了？
001000B0        beq     $1000be         * 終了ならend1へ
              † beq     end1
001000B4        bsr     $1000fc         * 次のシーンへコピーして
              † bsr     copy
001000B8        bsr     $1000dc         * 新しいページを表示
              † bsr     newpage
001000BC        rts
        end1:
001000BE        moveq   #$72.d0         * _ms_curoff
001000C0        trap    #15
001000C2        move.w  #$100+16.d1     * グラフィックONのまま768×512に
001000C6        moveq   #$10.d0         * _crtmod
001000C8        trap    #15
001000CA        moveq   #$ae.d0         * _os_curon
001000CC        trap    #15
001000CE        move.w  #0.-(sp)        * fncキー表示
001000D2        move.w  #14.-(sp)
001000D6        _conctrl
001000D8        addq.l  #4.sp
001000DA        _exit
        *
        *   新しいページをシーン番号(D7)に応じて表示
        *   D6にはシーン表示位置($000x_000y : x.y = 0～3)が入る
        *
        newpage:
001000DC        move.l  d7.d6           * シーン番号をD6に
001000DE        divu    #4.d6           * 4で割った余りと商を計算
001000E2        move.l  d6.d1           * d6をd1にコピー
001000E4        asl.l   #8.d1           * 256倍して座標に変換
001000E6        move.l  d1.$10017e
              † move.l  d1.xystart
001000EC        move.w  d1.d3           * d1.wはy座標
001000EE        swap    d1              * d1の上位ワードと下位ワードを交換
001000F0        move.w  d1.d2           * d1.wはx座標
001000F2        move.w  #1.d1           * ページ0をスクロール
001000F6        moveq   #$b3.d0         * home
001000F8        trap    #15
001000FA        rts
        *
        *   現在のシーンに描かれている絵を，
        *   輝度を落として次のシーンにコピーする
        *       in : d6 = 現在のシーンの画面上の位置($000x_000y)
        *            d7 = 次のシーンナンバー
        *
        copy:
001000FC        move.l  a1.-(sp)        * A1を保存
001000FE        bsr     $100160         * d0 = d6位置のGRAMオフセット
              † bsr     calc_ofst
00100102        movea.l d0.a2           * コピー元アドレス決定
00100104        move.l  d7.d6
00100106        divu    #4.d6
0010010A        bsr     $100160
```

```
          † bsr     calc_ofst
0010010E    movea.l d0,a3         * コピー先アドレス決定
00100110    movea.l #0,a1         * スーパーバイザへ
00100116    moveq   #$81,d0       * _b_super
00100118    trap    #15
0010011A    move.w  #255,d1
      copy1:
0010011E    move.w  #255,d2
      copy2:
00100122    move.w  (a2)+,d3      * ドットの色をD3に取り出す
00100124    bne     $10012e       * 黒でなければcopy3へ
          † bne     copy3
00100128    addq.l  #2,a3         * A3を次のドットへ進める
0010012A    bra     $100142       * copy5へ
          † bra     copy5
      copy3:
0010012E    subq.w  #1,d3         * 色を1つ小さくする
00100130    cmpi.w  #14,d3        * 灰色か?
00100134    beq     $100140       * 灰色ならcopy4へ
          † beq     copy4
00100138    move.w  #0,-2(a2)     * さもなければ灰色のドットを消去し
0010013E    moveq   #0,d3         * データを黒にする
      copy4:
00100140    move.w  d3,(a3)+      * 次のページに書き込む
      copy5:
00100142    dbra    d2,$100122    * 次のドットの処理
          † dbra    d2,copy2
00100146    adda.l  #768*2,a2     * 256ドット転送したら
0010014C    adda.l  #768*2,a3     * 次のラインのアドレスをセット
00100152    dbra    d1,$10011e    * 次のラインの処理
          † dbra    d1,copy1
00100156    movea.l d0,a1         * ユーザへ復帰
00100158    moveq   #$81,d0       * _b_super
0010015A    trap    #15
0010015C    movea.l (sp)+,a1      * A1を復帰
0010015E    rts
      *
      *   home座標のGRAMアドレスを求める
      *       in  : d6 = x位置×10000H +y位置
      *       out : d0 = GRAMアドレス
      *
      calc_ofst:
00100160    move.l  #1024*2,d1    * 1ラインオフセット
00100166    mulu    d6,d1
00100168    asl.l   #8,d1         * 256倍してy座標オフセット
0010016A    move.l  #256*2,d0     * x座標オフセット
00100170    swap    d6
00100172    mulu    d6,d0
00100174    add.l   d1,d0         * 先頭からのオフセット
00100176    addi.l  #$c00000,d0   * アドレス
0010017C    rts
      xystart:
0010017E    dc.w    0             * home用左上X座標
00100180    dc.w    0             * home用左上Y座標
      lndata:
00100182    dc.w    0             * 始点x座標
00100184    dc.w    0             * 始点y座標
00100186    dc.w    0             * 終点x座標
00100188    dc.w    0             * 終点y座標
0010018A    dc.w    15            * 色
0010018C    dc.w    $FFFF         * ラインスタイル
0010018E
```

クロールに必要なデータは，実画面のどの座標を表示画面の左上隅とするかである。これを算出するのに御仁が使った方法は次回のお楽しみとしておく。今回は機能だけを覚えておいていただきたい。

次にcopyサブルーチンだが，こちらはまずA2レジスタに現在表示している画面の左上アドレス，A3レジスタに次に表示する画面の左上アドレスを計算する。続いて，

move.w　(a2)＋, d3

でD3.wレジスタにドットの色を取り出し，それが白なら灰色に変換し，

move.w　d3,(a3)＋

で書き込むという作業を行っている。取り出したデータが灰色なら黒に変換して，取り出した場所，すなわち「－2(a2)」に書き込む。256ドット書き込めば1ライン分のデータ転送は終了し，次のラインのデータ転送にとりかかる。これを256ライン分行えば終了である。つまり，現在の画面に表示されている絵は次の画面では灰色で表示されるのだ。そして，灰色で表示されている部分はこの転送作業によって消去されることになる。

リスト最後のcalc_ofstサブルーチンは，転送アドレスを計算している部分である。これも次回のお楽しみということにしておこう。

## 絵を表示して眺める

絵を描くだけではどうしようもない。最後に御仁が作ったのが絵を表示して眺めるプログラムである。最初は単純に，描いた順に表示画面を変更していくだけのプログラムだったのだが，最終的に好きな順に表示できるように変更された。リスト3である。

このプログラムでは10007A$_H$のqueというラベルがつけられているアドレスから，表示する画面の番号がバイトサイズで並べられている。ここでは0番から15番までが順に並んでいるが，諸兄の描いた絵に合わせて好きなように並べていただきたい。御仁は画面の上からボールが落ちてくる絵を描き，それを順方向，逆方向に何度も表示するようにqueを設定して悦に入っている。これでボールが弾んでいるように見えるのである。queはFF$_H$，つまりバイトデータの－1で終了するようになっている。

プログラムは実に簡単である。queのアドレスをA0レジスタに保持しておき，

move.b　(a0)＋, d7

で画面番号を取り出す。取り出したデータが負の数なら終了，そうでなければリスト2のnewpageサブルーチンを使ってD7.lにセットされている番号の画面を表示するだけである。

表示してある絵をしばらく眺めるため，100032～10003A$_H$で時間稼ぎのループを形成している。稼げる時間は100032$_H$でD0.lレジスタにセットされている値によって変化する。大きくすれば長く，小さくすれば短くできるわけである。ここでは40000$_H$がセットされているが，これは御仁が経験的に決めた値である。諸兄の好みによって変更されたい。吾輩は10MHzの古式ゆかしいX68000なので，最新のXVIを使用している諸兄は，1.5倍した60000$_H$くらいがいいだろう。1枚の絵を表示する時間は決まっているので，同じ絵をしばらく表示し続けたい場合はqueに同じ画面番号を並べて対処されたい。

## プログラムと絵の保存

習作プログラム，描画プログラム，表示プログラムの3つのリストをお目に掛けてきたわけだが，実はリスト2のプログラムは完成品ではない。描いた線を消去する手段がないので絵を修正することができないし，前の画面が灰色で表示されているとはいえ，同じものを再び描

かなければならないというのも面倒なところである。

御仁は,「修正できないというのも,また味があっていい」などとほざいていたのだが,最終的にこれらの機能も盛り込むことに決めたようだ。また,指定した範囲の灰色のドットを白に変換する機能を用意し,前の絵の一部または全部をいま描いている絵に持ち込めるように変更した。それが最後に掲げる大きなリストである。

追加した機能を選択するためにポップアップメニューまで用意した力作だが,なにぶんにも初めての試みであるので命令の使い方が統一されていなかったり,無駄な命令を使っていたりといった力量不足の部分も目立つ。許してやっていただきたい。例によってデバッガ用のリストになっているが,アセンブラ用のリストとしても利用できる。諸兄の好みの方法で試されたい。

これだけのサイズのプログラムになると,ディスクに保存しておきたいと思うのは道理であろう。毎回デバッガでこのリストを入力するというのは考えものである。デバッガには,メモリのデータをディスクに保存しておくためのコマンド「W」が用意されている。これは,

　　－W ファイル名,開始アドレス　終了アドレス

として使う。リスト4を入力したら,

　　－w draw.dbx,100000 10065a

とでも入力すればいい。これを読み込むには開始アドレスを指定して,

　　－r draw.dbx,100000

とする。これでいつでもプログラムを使うことができるわけである。

描いたグラフィックも,このWコマンドで保存,Rコマンドで読み出すことができる。ただし,リスト4もリスト3も実画面モードを1024×1024としているので,単純に,

　　－w Xmas.dat,c00000 dfffff

などとして保存したのでは2Mバイトものファイルになってしまう。画面モードを512×512ドット,65536色モードに変換してから,

　　－w Xmas.dat,c00000 c7ffff

で保存していただきたい。画面モードの変換プログラムはいまさら挙げるまでもないだろう。諸兄は十分ご習熟なさっているはずである。もちろん65536色モードでAPICをお使いになっても構わない。いずれにしても,終了時に画面を65536色モードにするようプログラムを変更しておくと便利である。

＊

吾輩と御仁からのこころばかりのプレゼント。どうぞ使ってみていただきたい。

**リスト3　絵を眺める**

```
-an 100000
00100000        move.w  #3.-(sp)        * fncキー非表示
00100004        move.w  #14.-(sp)
00100008        _conctrl
0010000A        addq.l  #4.sp
0010000C        moveq   #$af.d0         * _os_curof
0010000E        trap    #15
00100010        move.b  #2.d1           * 画面消去
00100014        moveq   #$2a.d0         * _b_clr_st
00100016        trap    #15
00100018        move.w  #$100+2.d1      * 256×256ドット×16色×1
0010001C        moveq   #$10.d0         * _crtmod
0010001E        trap    #15
00100020        movea.l #$10007a.a0     * シーンナンバ配列
              † movea.l #que.a0
00100026        moveq   #0.d7           * d7をクリア
        loop:
00100028        move.b  (a0)+.d7        * シーンナンバを取り出す
0010002A        bmi     $100042         * 負の数なら終了
              † bmi     end
0010002E        bsr     $100060         * シーンナンバに応じた画面を表示
              † bsr     newpage
00100032        move.l  #$40000.d0      * 表示時間
        loop1:
00100038        subq.l  #1.d0
0010003A        bne     $100038
              † bne     loop1
0010003E        bra     $100028
              † bra     loop
        end:
00100042        moveq   #$72.d0         * _ms_curoff
00100044        trap    #15
00100046        move.w  #$100+16.d1     * グラフィックONのまま768×512に
0010004A        moveq   #$10.d0         * _crtmod
0010004C        trap    #15
0010004E        moveq   #$ae.d0         * _os_curon
00100050        trap    #15
00100052        move.w  #0.-(sp)        * fncキー表示
00100056        move.w  #14.-(sp)
0010005A        _conctrl
0010005C        addq.l  #4.sp
0010005E        _exit
        newpage:
00100060        move.l  d7.d6           * シーン番号をD6に
00100062        divu    #4.d6           * 4で割った余りと商を計算
00100066        move.l  d6.dl           * d6をdlにコピー
00100068        asl.l   #8.dl           * 256倍して座標に変換
0010006A        move.w  dl.d3           * dl.wはy座標
0010006C        swap    dl              * dlの上位ワードと下位ワードを交換
0010006E        move.w  dl.d2           * dl.wはx座標
00100070        move.w  #1.dl           * ページ0をスクロール
00100074        moveq   #$b3.d0         * home
00100076        trap    #15
00100078        rts
        que:
0010007A        dc.b    0.1.2.3.4.5.6.7
00100082        dc.b    8.9.10.11.12.13
00100088        dc.b    14.15
0010008A        dc.b    $ff
0010008C
```

**リスト4　描画プログラム（男は気合バージョン）**

```
-z0=100000
-an .z0
        †_exit          equ     $FF00
        †_conctrl       equ     $FF23
        †__ltos         equ     $FE11
00100000        move.w  #3,-(sp)        * fncキー非表示
00100004        move.w  #14,-(sp)
00100008        _conctrl
              † dc.w    _conctrl
0010000A        addq.l  #4,sp
0010000C        moveq   #$af,d0         * _os_curof
0010000E        trap    #15
00100010        move.w  #2,d1           * 256×256ドット×16色×1
00100014        moveq   #$10,d0         * _crtmod
00100016        trap    #15
00100018        moveq   #$90,d0         * _g_clr_on
0010001A        trap    #15
0010001C        move.w  #0,d1           * 表示ウィンドウ左上x座標
00100020        move.w  d1,d2           *                 y座標
00100022        move.w  #1023,d3        *             右下x座標
00100026        move.w  d3,d4           *                 y座標
00100028        moveq   #$b4,d0         * _window
0010002A        trap    #15
0010002C        move.l  #0,d1           * ソフトキーボード消去
00100032        moveq   #$7d,d0         * _skey_mod
00100034        trap    #15
00100036        moveq   #$70,d0         * _ms_init
00100038        trap    #15
0010003A        moveq   #$71,d0         * _ms_curon
0010003C        trap    #15
0010003E        move.l  #0,d1           * (0,0)
00100044        move.l  #$ff00ff,d2     * (255,255)
0010004A        moveq   #$77,d0         * _ms_limit
0010004C        trap    #15
0010004E        movea.l #$1003ba,a1     * line用データアドレス
              † movea.l #lndata,a1
```

```
00100054          moveq    #0,d7               * d7はシーンナンバ
00100056          bsr      $100310             * ナンバに応じたページを表示
                † bsr      newpage
        loop:
0010005A          moveq    #$74,d0             * _ms_getdt
0010005C          trap     #15
0010005E          tst.b    d0                  * 右ボタンが
00100060          bmi      $100074             * 押された
                † bmi      loop1
00100064          tst.w    d0                  * 左ボタンが
00100066          bmi      $10007c             * 押された
                † bmi      draw
0010006A          move.l   #-1,(a1)
00100070          bra      $10005a
                † bra      loop
        loop1:
00100074          bsr      $1000a4
                † bsr      popup
00100078          bra      $10005a
                † bra      loop
        draw:
0010007C          moveq    #$75,d0             * _ms_curgt
0010007E          trap     #15
00100080          add.l    $1003b2,d0
                † add.l    xystart,d0
00100086          move.l   d0,4(a1)            * xy座標セット
0010008A          tst.w    (a1)                * 始点x座標が
0010008C          bpl      $100094             * 正の数だったら描画
                † bpl      draw1
00100090          move.l   4(a1),(a1)          * さもなければ終点座標をコピー
        draw1:
00100094          movea.l  $1003b6,a0          * Pset or Reset or Copy
                † movea.l  drawmode,a0
0010009A          jsr      (a0)
0010009C          move.l   4(a1),(a1)
001000A0          bra      $10005a
                † bra      loop
        *
        *   ポップアップメニュー
        *
        popup:
001000A4          movea.l  #$1000b0,a0
                † movea.l  #menu,a0
001000AA          bsr.w    $1003c6
                † bsr.w    make_popup
001000AE          rts
        menu:
001000B0          dc.w     6*7+12              * メニュー横幅
001000B2          dc.w     14*6+12             * メニュー縦幅
001000B4          dc.l     $1000e8,0           * ラベルと処理ルーチンアドレス
001000BC          dc.l     $1000f0,$100110
001000C4          dc.l     $1000f6,$100142
001000CC          dc.l     $1000fe,$100192
001000D4          dc.l     $100104,$1002a2
001000DC          dc.l     $10010a,$1002b6
                † dc.l     item0,0             * ラベルと処理ルーチンアドレス
                † dc.l     item1,do_pset
                † dc.l     item2,do_erase
                † dc.l     item3,do_copy
                † dc.l     item4,do_next
                † dc.l     item5,next1
001000E4          dc.l     0                   * エンドマーク
        item0:
001000E8          dc.b     "=No.  =",0
        item1:
001000F0          dc.b     "*Pset",0
        item2:
001000F6          dc.b     " Erase",0
        item3:
001000FE          dc.b     " Copy",0
        item4:
00100104          dc.b     " Next",0
        item5:
0010010A          dc.b     " End",0
                † .even
        do_pset:
00100110          move.b   #'*',$1000f0        * psetをチェック
00100118          move.b   #' ',$1000f6
00100120          move.b   #' ',$1000fe
                † move.b   #'*',item1
                † move.b   #' ',item2
                † move.b   #' ',item3
00100128          move.w   #15,$1003ba+8       * 色を白に
                † move.w   #15,lndata+8
00100130          move.l   #$10013c,$1003b6
                † move.l   #do_pset1,drawmode
0010013A          rts
        do_pset1:
0010013C          moveq    #$b8,d0             * _line
0010013E          trap     #15
00100140          rts
        do_erase:
00100142          move.b   #' ',$1000f0
0010014A          move.b   #'*',$1000f6        * resetをチェック
00100152          move.b   #' ',$1000fe
                † move.b   #' ',item1
                † move.b   #'*',item2
                † move.b   #' ',item3
0010015A          move.w   #0,$1003ba+8        * 色を黒に
                † move.w   #0,lndata+8
00100162          move.l   #$10016e,$1003b6
                † move.l   #do_erase1,drawmode
0010016C          rts
        do_erase1:
0010016E          move.l   $1003ba+4,-(sp) * 終点座標保存
                † move.l   lndata+4,-(sp)
00100174          move.l   $1003ba,d0
                † move.l   lndata,d0
0010017A          addi.l   #$00080008,d0       * (8,8)
00100180          move.l   d0,$1003ba+4
                † move.l   d0,lndata+4
00100186          moveq    #$ba,d0             * _fill
00100188          trap     #15
0010018A          move.l   (sp)+,$1003ba+4 * 終点座標復帰
                † move.l   (sp)+,lndata+4
00100190          rts
        do_copy:
00100192          move.b   #' ',$1000f0
0010019A          move.b   #' ',$1000f6
001001A2          move.b   #'*',$1000fe        * copyをチェック
                † move.b   #' ',item1
                † move.b   #' ',item2
                † move.b   #'*',item3
001001AA          move.l   #$1001b6,$1003b6
                † move.l   #do_copy1,drawmode
001001B4          rts
        do_copy1:
001001B6          move.l   #0,d2               * XY方向の長さは0
001001BC          move.l   d2,$1005ac+6
                † move.l   d2,region+6
001001C2          move.l   d2,d1               * 最初Boxはoff
001001C4          move.l   $1003ba,d0                  * 始点取り出し
                † move.l   lndata,d0
001001CA          sub.l    $1003b2,d0          * テキスト座標に変換
                † sub.l    xystart,d0
001001D0          move.l   d0,$1005ac+2        * 始点セット
                † move.l   d0,region+2
        do_copy2:
001001D6          moveq    #$74,d0             * _ms_getdt
001001D8          trap     #15
001001DA          tst.w    d0                  * 左ボタンが
001001DC          bpl      $100222             * 離されればdo_copy4へ
                † bpl      do_copy4
001001E0          moveq    #$75,d0             * _ms_curgt
001001E2          trap     #15
001001E4          cmp.w    $1005ac+4,d0        * Y座標チェック
                † cmp.w    region+4,d0
001001EA          blt      $100210             * 始点>D0.wならdo_copy3へ
                † blt      do_copy3
001001EE          sub.l    $1005ac+2,d0        * XY座標チェック
                † sub.l    region+2,d0
001001F4          blt      $100210             * 始点>D0ならdo_copy3
                † blt      do_copy3
001001F8          addi.l   #$00010001,d0       * XY座標を1つ大きくする
001001FE          cmp.l    d2,d0               * 前のxylenと比較
00100200          beq      $1001d6             * 同じなら描画しない
                † beq      do_copy2
00100204          move.l   d0,d2               * 違えばD2を更新
00100206          bsr      $10057a             * 四角を表示
                † bsr      drawregion
0010020A          moveq    #1,d1               * Box表示on
0010020C          bra      $1001d6
                † bra      do_copy2
        do_copy3:
00100210          tst.l    d1                  * Box表示onか
00100212          beq      $1001d6             * offならdo_copy2へ
                † beq      do_copy2
00100216          moveq    #0,d0               * 長さを0にして
00100218          bsr      $10057a             * 四角を表示
                † bsr      drawregion
0010021C          moveq    #0,d1               * Box表示off
0010021E          bra      $1001d6
                † bra      do_copy2
        do_copy4:
00100222          tst.w    d2                  * Y幅が
00100224          beq      $10029a             * 0ならdo_copy8へ
                † beq      do_copy8
00100228          swap     d2
0010022A          tst.w    d2                  * X幅が
0010022C          beq      $10029a             * 0ならdo_copy8へ
                † beq      do_copy8
00100230          swap     d2
00100232          movem.l  d6/a1,-(sp)         * D6/A1を保存
00100236          movea.l  #0,a1
0010023C          moveq    #$81,d0             * _b_super
0010023E          trap     #15
00100240          movea.l  d0,a1               * uspをセット
00100242          bsr      $100394             * GRAMオフセットを計算
                † bsr      calc_ofst
00100246          movea.l  d0,a0               * A0にオフセットをコピー
00100248          moveq    #0,d0               * D0をクリア
0010024A          move.w   $1005ac+2,d0        * 始点座標をD0.wに取り出す
                † move.w   region+2,d0
00100250          asl.l    #1,d0               * 2倍してXオフセットを計算
00100252          adda.l   d0,a0               * A0に加える
00100254          move.w   $1005ac+4,d1        * Y座標をD1.wに
                † move.w   region+4,d1
0010025A          mulu     #1024*2,d1          * Yオフセットを計算
0010025E          adda.l   d1,a0               * A0=始点アドレス
00100260          move.w   d2,d3               * Y幅をD3.wにセット
00100262          swap     d2                  * X幅をD2.wにセット
00100264          moveq    #0,d1               * D1をクリア
00100266          move.w   d2,d1               * X幅をD1にセット
00100268          neg.l    d1                  * 符号反転
0010026A          add.l    #1024,d1            * 次のラインへのドット数
00100270          asl.l    #1,d1               * 次のラインへのバイト数
00100272          subq.l   #1,d2               * X方向転送回数
00100274          move.l   d2,d4               * D4に保存
00100276          subq.l   #1,d3               * Y方向転送回数
        do_copy5:
00100278          move.l   d4,d2               * X方向転送回数
        do_copy6:
0010027A          cmpi.w   #14,(a0)+           * 灰色か
0010027E          bne      $100288             * 違えばdo_copy7へ
                † bne      do_copy7
00100282          move.w   #15,-2(a0)          * 色を白に
        do_copy7:
00100288          dbra     d2,$10027a
                † dbra     d2,do_copy6
0010028C          adda.l   d1,a0               * 次のラインへ
0010028E          dbra     d3,$100278
                † dbra     d3,do_copy5
00100292          moveq    #$81,d0             * _b_super
00100294          trap     #15
```

▶「ようこそここへC言語」が最終回ということで残念です。C関連の記事は今後とも続けていってほしいです。
松川　努(25)兵庫県

```
00100296        movem.l (sp)+,d6/a1     * D6/A1を復帰
         do_copy8:
0010029A        moveq   #0,d0           * 長さを0にして
0010029C        bsr     $10057a         * 四角を表示
              † bsr     drawregion
001002A0        rts
         do_next:
001002A2        addq.l  #1,d7           * 次のシーンへ
001002A4        cmpi.w  #16,d7          * シーン15まで終了？
001002A8        beq     $1002b6         * 終了ならnext1へ
              † beq     next1
001002AC        bsr     $100330         * 次のシーンへコピーして
              † bsr     copy
001002B0        bsr     $100310         * 新しいページを表示
              † bsr     newpage
001002B4        rts
         next1:
001002B6        bsr     $100394
              † bsr     calc_ofst
001002BA        movea.l d0,a2           * 最後のページの先頭アドレス
001002BC        movea.l #0,a1           * スーパーバイザへ
001002C2        moveq   #$81,d0         * _b_super
001002C4        trap    #15
001002C6        move.w  #255,d1
         next2:
001002CA        move.w  #255,d2
         next3:
001002CE        move.w  (a2)+,d3        * ドットの色をD3に取り出す
001002D0        cmpi.w  #14,d3          * 灰色かどうかチェック
001002D4        bne     $1002de         * 灰色以外ならnext4へ
              † bne     next4
001002D8        move.w  #0,-2(a2)       * 灰色なら黒にする
         next4:
001002DE        dbra    d2,$1002ce
              † dbra    d2,next3
001002E2        adda.l  #768*2,a2       * 次のラインのアドレスをセット
001002E8        dbra    d1,$1002ca
              † dbra    d1,next2
001002EC        movea.l d0,a1           * ユーザへ復帰
001002EE        moveq   #$81,d0         * _b_super
001002F0        trap    #15
001002F2        moveq   #$72,d0         * _ms_curoff
001002F4        trap    #15
001002F6        move.w  #$100+16,d1     * グラフィックONのまま768×512に
001002FA        moveq   #$10,d0         * _crtmod
001002FC        trap    #15
001002FE        moveq   #$ae,d0         * _os_curon
00100300        trap    #15
00100302        move.w  #0,-(sp)        * fncキー表示
00100306        move.w  #14,-(sp)
0010030A        _conctrl
              † dc.w    _conctrl
0010030C        addq.l  #4,sp
0010030E        _exit
              † dc.w    _exit
         *
         *      新しいページをシーン番号(D7)に応じて表示
         *      D6にはシーン表示位置($000x_000y : x,y = 0～3)が入る
         *
         newpage:
00100310        move.l  d7,d6           * シーン番号をD6に
00100312        divu    #4,d6           * 4で割った余りと商を計算
00100316        move.l  d6,d1           * d6をd1にコピー
00100318        asl.l   #8,d1           * 256倍して座標に変換
0010031A        move.l  d1,$1003b2
              † move.l  d1,xystart
00100320        move.w  d1,d3           * d1.wはy座標
00100322        swap    d1              * d1の上位ワードと下位ワードを交換
00100324        move.w  d1,d2           * d1.wはx座標
00100326        move.w  #1,d1           * ページ0をスクロール
0010032A        moveq   #$b3,d0         * _home
0010032C        trap    #15
0010032E        rts
         *
         *      現在のシーンに描かれている絵を,
         *      輝度を落として次のシーンにコピーする
         *          in  : d6 = 現在のシーンの画面上の位置($000x_000y)
         *                d7 = 次のシーンナンバー
         *
         copy:
00100330        move.l  a1,-(sp)        * A1を保存
00100332        bsr     $100394         * d0 = d6座標のGRAMオフセット
              † bsr     calc_ofst
00100336        movea.l d0,a2           * コピー元アドレス決定
00100338        move.l  d7,d6
0010033A        divu    #4,d6
0010033E        bsr     $100394
              † bsr     calc_ofst
00100342        movea.l d0,a3           * コピー先アドレス決定
00100344        movea.l #0,a1           * スーパーバイザへ
0010034A        moveq   #$81,d0         * _b_super
0010034C        trap    #15
0010034E        move.w  #255,d1
         copy1:
00100352        move.w  #255,d2
         copy2:
00100356        move.w  (a2)+,d3        * ドットの色をD3に取り出す
00100358        bne     $100362         * 黒でなければcopy3へ
              † bne     copy3
0010035C        addq.l  #2,a3           * A3を次のドットへ進める
0010035E        bra     $100376         * copy5へ
              † bra     copy5
         copy3:
00100362        subq.w  #1,d3           * 色を1つ小さくする
00100364        cmpi.w  #14,d3          * 灰色か？
00100368        beq     $100374         * 灰色ならcopy4へ
              † beq     copy4
0010036C        move.w  #0,-2(a2)       * さもなければ灰色のドットを消去し
00100372        moveq   #0,d3           * データを黒にする
         copy4:
00100374        move.w  d3,(a3)+        * 次のページに書き込む
         copy5:
```

```
00100376        dbra    d2,$100356      * 次のドットの処理
              † dbra    d2,copy2
0010037A        adda.l  #768*2,a2       * 256ドット転送したら
00100380        adda.l  #768*2,a3       * 次のラインのアドレスをセット
00100386        dbra    d1,$100352      * 次のラインの処理
              † dbra    d1,copy1
0010038A        movea.l d0,a1           * ユーザへ復帰
0010038C        moveq   #$81,d0         * _b_super
0010038E        trap    #15
00100390        movea.l (sp)+,a1        * A1を復帰
00100392        rts
         *
         *      home座標のGRAM先頭からのオフセットを求める
         *          in  : d6 = x座標×10000H+y座標
         *          out : d0 = オフセット
         *
         calc_ofst:
00100394        move.l  #1024*2,d1      * 1ラインオフセット
0010039A        mulu    d6,d1
0010039C        asl.l   #8,d1           * 256倍してy座標オフセット
0010039E        move.l  #256*2,d0       * x座標オフセット
001003A4        swap    d6
001003A6        mulu    d6,d0
001003A8        add.l   d1,d0           * 先頭からのオフセット
001003AA        addi.l  #$c00000,d0
001003B0        rts
         xystart:
001003B2        dc.w    0               * home用左上X座標
001003B4        dc.w    0               * home用左上Y座標
         drawmode:
001003B6        dc.l    $10013c
              † dc.l    do_pset1
         lndata:
001003BA        dc.w    0               * 始点x座標
001003BC        dc.w    0               * 始点y座標
001003BE        dc.w    0               * 終点x座標
001003C0        dc.w    0               * 終点y座標
001003C2        dc.w    15              * 色
001003C4        dc.w    $FFFF           * ラインスタイル
         *
         *      ポップアップメニューを作る
         *      in : A0 = メニューアイテムアドレス
         *      break : d0/a0
         *
              † .even
         make_popup:
001003C6        movem.l d1/d2/a1/a2,-(sp)
001003CA        movea.l a0,a1           * A0を保存
001003CC        move.l  d7,d0           * シーン番号を
001003CE        addq.l  #1,d0
001003D0        lea     $1000e8+4,a0    * メニューに
              † lea     item0+4,a0
001003D6        cmpi.w  #9,d0
001003DA        bgt     $1003e0
              † bgt     m_str1
001003DE        addq.l  #1,a0
         m_str1:
001003E0        __ltos                  * セットする
              † dc.w    __ltos
001003E2        move.b  #'=',$1000e8+6  * =を補う
              † move.b  #'=',item0+6
001003EA        movea.l a1,a0           * A0復帰
001003EC        movea.l a0,a2           * menu_adrs
001003EE        moveq   #$75,d0         * _ms_curgt
001003F0        trap    #15
001003F2        move.l  d0,d1           * マウス座標
001003F4        moveq   #$72,d0         * _ms_curof
001003F6        trap    #15
001003F8        cmpi.w  #128,d1         * 画面の半分より
001003FC        bmi.w   $100404         * 上ならm_pop1へ
              † bmi.w   m_pop1
00100400        sub.w   2(a2),d1        * メニューの縦幅を引く
         m_pop1:
00100404        swap    d1              * xy座標交換
00100406        cmpi.w  #128,d1         * 画面の半分より
0010040A        bmi.w   $100410         * 右ならm_pop2へ
              † bmi.w   m_pop2
0010040E        sub.w   (a2),d1         * メニューの横幅を引く
         m_pop2:
00100410        swap    d1              * xy座標交換
00100412        move.w  #2,$1005b8      * プレーン2を指定
              † move.w  #2,boxData
0010041A        move.l  d1,$1005b8+2    * 左上座標格納
         move.l  d1,boxData+2
00100420        move.l  (a2),$1005b8+6  * メニュー幅格納
              † move.l  (a2),boxData+6
00100426        move.w  #$FFFF,$1005b8+10       * スタイル格納
              † move.w  #$FFFF,boxData+10
0010042E        lea     $1005b8,a1      * boxfillデータアドレスセット
              † lea     boxData,a1
00100434        moveq   #$d7,d0         * _txfill
00100436        trap    #15
00100438        move.l  $1005b8+2,d1    * 文字表示座標を決定
              † move.l  boxData+2,d1
0010043E        addi.l  #$00060006,d1   * (x+6,y+6)
00100444        lea     4(a2),a1        * アイテムの先頭座標
         m_pop3:
00100448        tst.l   (a1)            * 終端か
0010044A        beq.w   $100462         * そうならメニュー選択
              † beq.w   m_pop4
0010044E        movea.l (a1),a0         * メニュー文字列を取り出す
00100450        move.l  d1,d0           * 座標セット
00100452        bsr.w   $1005d4
              † bsr.w   putstr
00100456        addq.l  #8,a1           * 次の文字へ
00100458        addi.l  #14,d1          * 13ドット改行
0010045E        bra.w   $100448
              † bra.w   m_pop3
         m_pop4:
00100462        move.w  $1005b8+4,d0    * メニュー表示Y座標
              † move.w  boxData+4,d0
00100468        move.w  d0,d1
```

▶まさか今頃になってSPSの迷作「フルスロットル」が載るとは（付録17ページ）。
岡部　祥明(17)福島県

```
0010046A          add.w    #20,d1
0010046E          move.w   d1,$1005c4        * 最初のアイテムのY座標
                † move.w   d1,menuBorder
00100474          add.w    $1005b8+8,d0
                † add.w    boxData+8,d0
0010047A          subq.w   #7,d0
0010047C          move.w   d0,$1005c4+2      * 最後のアイテムY座標
                † move.w   d0,menuBorder+2
00100482          lea      $1005c8,a1        * アイテム反転用意
                † lea      menuBox,a1
00100488          move.w   #3,(a1)           * plane
0010048C          move.w   $1005b8+2,d0      * xpos
                † move.w   boxData+2,d0
00100492          addq.w   #3,d0
00100494          move.w   d0,2(a1)
00100498          move.w   $1005b8+6,d0      * xlen
                † move.w   boxData+6,d0
0010049E          subq.w   #6,d0
001004A0          move.w   d0,6(a1)
001004A4          move.w   #14,8(a1)         * ylen
001004AA          move.w   #$FFFF,10(a1)     * style
001004B0          lea      $1005c4,a0        * ボーダ設定
                † lea      menuBorder,a0
001004B6          bsr.w    $100504
                † bsr.w    select
001004BA          moveq    #$72,d0           * _ms_curof
001004BC          trap     #15
001004BE          move.w   #2,$1005b8
                † move.w   #2,boxData
001004C6          move.w   #0,$1005b8+10
                † move.w   #0,boxData+10
001004CE          lea      $1005b8,a1
                † lea      boxData,a1
001004D4          moveq    #$d7,d0           * _txfill
001004D6          trap     #15
001004D8          move.w   #3,$1005b8
                † move.w   #3,boxData
001004E0          moveq    #$d7,d0           * _txfill
001004E2          trap     #15
001004E4          moveq    #$71,d0           * _ms_curon
001004E6          trap     #15
001004E8          tst.l    d1                * アイテムが
001004EA          bmi      $1004fe           * 選択されていない
                † bmi      m_pop5
001004EE          asl.l    #3,d1             * D1を8倍する
001004F0          tst.l    16(a2,d1)         * 処理ルーチンをチェック
001004F4          beq      $1004fe           * 登録されていない
                † beq      m_pop5
001004F8          movea.l  16(a2,d1),a0
001004FC          jsr      (a0)
        m_pop5:
001004FE          movem.l  (sp)+,d1/d2/a1/a2
00100502          rts
        *
        *   メニュー選択
        *   in  : a0 = メニューボーダ
        *         ・a1 = 反転用ボックスデータ
        *   out : d1 = 選択された項目
        *
        select:
00100504          moveq    #$71,d0           * _ms_curon
00100506          trap     #15
00100508          moveq    #-1,d2            * 最初はどこも反転していない
        select1:
0010050A          moveq    #-1,d1            * 次に反転させるアイテム
0010050C          moveq    #$75,d0           * _ms_curgt
0010050E          trap     #15
00100510          cmp.w    2(a0),d0          * y-endより大きいか
00100514          bpl.w    $100530
                † bpl.w    select2
00100518          sub.w    (a0),d0           * y┗startを引く
0010051A          bmi.w    $100530
                † bmi.w    select2
0010051E          andi.l   #$FFFF,d0         * x座標をクリア
00100524          divu     #14,d0            * y座標を14で割る
00100528          andi.l   #$FFFF,d0         * 余りをクリア
0010052E          move.l   d0,d1             * 次に反転させるアイテム
        select2:
00100530          bsr.w    $100540           * アイテム反転
                † bsr.w    reverse
00100534          moveq    #$74,d0           * _ms_getdt
00100536          trap     #15
00100538          tst.b    d0
0010053A          bmi.w    $10050a           * 右ボタンONならループ
                † bmi.w    select1
0010053E          rts
        reverse:
00100540          cmp.l    d1,d2             * 同じアイテムなら終了
00100542          beq.w    $10055e
                † beq.w    reverse2
00100546          tst.l    d2
00100548          bmi.w    $100552           * 前回はメニュー外
                † bmi.w    reverse1
0010054C          move.l   d2,d0             * 前回のアイテム反転
0010054E          bsr.w    $100562
                † bsr.w    rev
        reverse1:
00100552          tst.l    d1
00100554          bmi.w    $10055e           * 今回はメニュー外
                † bmi.w    reverse2
00100558          move.l   d1,d0             * 今回のアイテム反転
0010055A          bsr.w    $100562
                † bsr.w    rev
        reverse2:
0010055E          move.l   d1,d2             * アイテム番号更新
00100560          rts
        rev:
00100562          mulu     #14,d0            * アイテム番号を14倍
00100566          add.w    (a0),d0           * y-startを加える
00100568          move.w   d0,4(a1)          * yposにセット
0010056C          moveq    #$72,d0           * _ms_curof
0010056E          trap     #15
00100570          moveq    #$d8,d0           * _txrev
00100572          trap     #15
00100574          moveq    #$71,d0           * _ms_curon
00100576          trap     #15
00100578          rts
        *
        *   Copy用の四角を表示する
        *   in : d0 = $xlen_ylen
        *
        drawregion:
0010057A          movem.l  d1/a1,-(sp)       * D1/A1を保存
0010057E          move.l   d0,d1             * D0を保存
00100580          moveq    #$72,d0           * _ms_curof
00100582          trap     #15
00100584          movea.l  #$1005ac,a1       * BOXデータアドレスをセット
                † movea.l  #region,a1
0010058A          move.w   #0,10(a1)         * 黒のラインで
00100590          moveq    #$d6,d0           * _txbox
00100592          trap     #15
00100594          move.l   d1,6(a1)          * xylenをセットし
00100598          move.w   #$FFFF,10(a1)     * 白のラインで
0010059E          moveq    #$d6,d0           * _txbox
001005A0          trap     #15
001005A2          moveq    #$71,d0           * _ms_curon
001005A4          trap     #15
001005A6          movem.l  (sp)+,d1/a1       * D1/A1を復帰
001005AA          rts
        region:
001005AC          dc.w     2                 * テキストプレーン2
001005AE          dc.w     0,0               * 始点XY座標
001005B2          dc.w     0,0               * XY方向の長さ
001005B6          dc.w     $FFFF             * ラインスタイル
        boxData:
001005B8          dc.w     0                 * plane
001005BA          dc.w     0                 * xpos
001005BC          dc.w     0                 * ypos
001005BE          dc.w     0                 * xlen
001005C0          dc.w     0                 * ylen
001005C2          dc.w     0                 * style
        menuBorder:
001005C4          dc.w     0                 * y-start
001005C6          dc.w     0                 * y-end
        menuBox:
001005C8          dc.w     0                 * plane
001005CA          dc.w     0                 * xpos
001005CC          dc.w     0                 * ypos
001005CE          dc.w     0                 * xlen
001005D0          dc.w     0                 * ylen
001005D2          dc.w     0                 * style
        *
        *   文字を表示する
        *   in : d0 = $xxxx_yyyyy 座標
        *          a0 = 文字列アドレス
        *   break : d0/a0
        *
                † .even
        putstr:
001005D4          movem.l  d1/a1,-(sp)
001005D8          move.l   d1,$10063a
                † move.l   d1,strpos
001005DE          move.b   #8,d1             * text plane 4
001005E2          moveq    #$15,d0           * _tcolor
001005E4          trap     #15
        putstr1:
001005E6          moveq    #0,d1
001005E8          move.b   (a0)+,d1
001005EA          beq.w    $10062c
                † beq.w    putstr3
001005EE          bpl.w    $1005f6
                † bpl.w    putstr2
001005F2          asl.l    #8,d1
001005F4          move.b   (a0)+,d1
        putstr2:
001005F6          addi.l   #6*$10000,d1
001005FC          movea.l  #$10063e,a1
                † movea.l  #chrbuf,a1
00100602          moveq    #$19,d0           * _fntget
00100604          trap     #15
00100606          move.w   $10063a,d1
                † move.w   strpos,d1
0010060C          move.w   $10063a+2,d2
                † move.w   strpos+2,d2
00100612          moveq    #$1b,d0           * _textput
00100614          trap     #15
00100616          move.w   $10063a,d0
                † move.w   strpos,d0
0010061C          add.w    $10063e,d0
                † add.w    chrbuf,d0
00100622          move.w   d0,$10063a
                † move.w   d0,strpos
00100628          bra.w    $1005e6
                † bra.w    putstr1
        putstr3:
0010062C          move.b   #1,d1             * text plane 1
00100630          moveq    #$15,d0           * _tcolor
00100632          trap     #15
00100634          movem.l  (sp)+,d1/a1
00100638          rts
        strpos:
0010063A          dc.w     0,0               * 文字表示座標
        chrbuf:
0010063E          dc.w     0                 * xdots
00100640          dc.w     0                 * ydots
00100642          dc.w     0,0,0,0,0,0       * font
0010064E          dc.w     0,0,0,0,0,0
0010065A
```

[特集]

SX-WINDOWの未来

# SX-WINDOW

ウィンドウ環境。X68000の生まれたときには誰もが期待していたものだ。限りなく皆無に近い環境から5年，現状を見ると我々はすでにパソコンレベルでは限界に近い環境を作り出しているといえる。
そしてSX-WINDOWから2年。多大な可能性を秘めたままSX-WINDOWは眠っている。ウィンドウ環境への移行はこれまでの資産を捨て去ることなのか？　本当にいま以上の環境が生まれるというのだろうか？　いい尽くされたことを並べてでも，もう一度確認しなければならないことがある。SXシステムが目指さなければならない高みにはまだ誰も到達していない。既存の環境を吸収しているだけでは未来は見えないのだから。

CONTENTS

どうしてウィンドウなのか？

# ウィンドウ環境のために

Nakano Shuichi 中野 修一

重い処理，ハードウェア資源を湯水のように消費するウィンドウシステム。にもかかわらず，どうしてウィンドウ環境が注目されるのでしょうか？　ここではウィンドウの果たす意味と課題について考えてみましょう。

## ウィンドウの必然性

ウィンドウといえば，いまやMS-WINDOWSによるウィンドウ環境がもっぱらの話題になっています。最初に少しだけWINDOWSの話をしましょう。

「GUI（グラフィカルユーザーインタフェイス）だから使いやすい」というところにウィンドウ環境の意義を求めるのは大きな誤解です。単なるGUIなら，すでにアプリケーションが独自に実現していることも多く，アプリケーションユースが中心のMS-DOSユーザーにとってさほど問題になるとは思えません。

MS-DOSの上に乗っかっているとはいえ，「80286/80386のプロテクトモードでアプリケーションを動かせるOS」ということにWINDOWSの価値があります（WINDONSシステム単体なら8088でも動作させることができる)。

X68000での環境を考えてみればすぐにおわかりでしょうが，メモリが640Kバイトしか使えない状態で「環境」と呼べるものを構築することがどれくらい大変かを考えるとプロテクトモードを使用できるWINDOWSのありがたみが想像できると思います。

PC-9801でWINDOWS ver.3.0がリリースされてもっとも話題になったのは「複数のアプリケーションを切り替えて実行できる」ということでした。これはPC-9801では画面が狭すぎてマルチウィンドウどころではないという事情が大きく作用していました。IBMの人は「あれは窓ではなくてドアだ」といったそうです。日本電気の人はたいそう怒ったらしいですが，アプリケーションの切り替えしか使い物にならない状況を実に適切に表現していました。

WINDOWS対応のアプリケーションが出始めてようやく状況も変わり始めています。しかしPC-9801では高解像度化が必須事項として挙げられているのも事実です。「初心者にも扱いやすいGUI環境」というのはひとまず置いておきましょう。ここではウィンドウ環境がもたらすものとそのための条件について考えてみましょう。

## マルチウィンドウへの適応

もう数年前のことになります。当時続々登場するマルチウィンドウ・マルチタスク環境というものを目にしても，なにか不完全さを感じることが多かったのです。1＋1を2にしかできないのなら2台のコンピュータを並べるのとどう違うのか，1＋1を3にも4にもしてこそ，マルチウィンドウ・マルチタスクの意味があるのではないか？　という直感によるものです。

システムを作る側にもそれを使う側にもなにをすべきかが理解されていないようなのです。そういったことを話していて，飛び出したのは「人間はまだマルチウィンドウを使いこなすほど進化していないのではないか？」という疑問でした。

とある雑誌に，コンピュータを使うことに対して「完璧な記憶力さえあればこんなものは必要ない」という論調のものがありました。その人はコンピュータやワープロを単なるメモとしてしか使っていなかったのでしょう。

たとえば，キーボードをある程度打てる人とそうでない人ではワープロなどはまったく違ったものとして受け取られます。思考を指先から画面に伝えることができる人とまとまった思考を1文字ずつ翻訳しながら打ち込む人では生産性以外になにか違いがあります。ワープロに慣れてくると，アイデアプロセッサとしての使い方,「ワープロ流」の思考法を身につけるようになるからです。

昔は原稿用紙に書いていた文章をワープロで清書するなどしていたのですが，最近ではもはやワープロなしでは文章は書けなくなりました。思考の手順自体をワープロが拡大してしまったからです。いまでは原稿用紙の端から順番に完成した文章を埋めていくことが神業のように思えます。

新しいツールと出合うことによって人間は能力を拡大できる，というとなにか大袈裟に聞こえますが，文明の発展もしくは人間の歴史というのは人間の身体的特徴が進化したことを示すのではなく，どんな道具を作ってきたかを示しているのを思い起こせばきわめて自然なものだとわかるでしょう。

これを「進化」というのは問題がありそうですからここでは「適応」と表現しましょう。

「まだウィンドウという環境に適応できていない」とすると，ウィンドウ環境はこれまでのパソコン環境とどう違うのでしょうか。

キーワードはマルチウィンドウ・マルチタスクです。マルチウィンドウとはすなわち「画面の情報量が多い」ことです。さらにマルチタスクを行うことは，ユーザーにとっては一度に複数台のコンピュータを扱うのと同じことになります。そこでは従来のパソコン環境が並列しているわけではなく，新しい次元で調整された環境が広がっています。1台のコンピュータで動いているということは，従来なら高度なネットワーク環境でしかできなかったようなことも(それ以上のことも)できるようになるということをも意味します（もっともそれだけメモリやCPUパワーも必要ということですが)。

そういった環境ではいったいどんなことが可能なのかを推し量るだけでも大変です。しかし，GUIを伴ったマルチウィンドウ環境がもたらす目眩にそろそろ適応していくことが必要ではないでしょうか。そこには，単に初心者に使いやすいこと（それ自体も非常に重要なことですが）以外に大きな可能性が広がっているのですから。

## SX-WINDOWの進路

では，どんなことが可能なのかということが問題になってきます。

複数のものが同時に使えるだけでもかなり環境は変わります。ウィンドウの機能や操作体系には多くの考え方があります。どういうものがいいというのは，まだ模索状況ですから，これまでにわかっているSXの作法から演繹していきましょう。新しいOS/2のプレゼンテーションマネージャはSX-WINDOWの操作法と似ているそうですから，SXの作法もそう悪いものではないでしょう（新しいだけあって基本部分は相当洗練されていると思う)。

実装上の問題をユーザー側からシミュレートしてみましょう。

想像してください。ウィンドウ上にテキストエディタが開いています。隣にはコンパイラのウィンドウも見えます。ソースプログラムを書き終えて，コンパイルに移るときどんな操作が望ましいのでしょうか？

現状を元に考えれば，テキストをセーブしそのファイルアイコンをコンパイラウィンドウへ放り込むことになるでしょう。わかりやすい代わりに少々うっとうしい感じもあります。

emacsなどではセーブしたテキストを子プロセス起動したコンパイラに渡してエラー時にはタグファイルを持って帰るということをまとめて行わせることもできます。これと同様なものは可能でしょうか？　無論，可能です。しかし，よく考えるとずいぶん無駄が多いことに気づきます。

隣にウィンドウが開いているのにコマンドを実行するのはメモリとディスクアクセスの無駄です。エディタ上にあるテキストをコンパイラがもう一度読み込むのはさらに無駄です。

エディタがタスク間通信でテキストをコンパイラに送れば状況はかなり改善されます。しかし，メモリは膨大に必要です。もっといいのはタスク間通信でデータエリアのメモリハンドルを渡すことです。もっとも，この場合はコンパイラがエディタのデータ管理法を知っている必要がありますが。

普通のコンパイラでさえメモリ2Mバイトでないと動かないとか，ウィンドウ自体もメモリ2Mバイト必要という事実はとりあえず無視しましょう。理想をいえば，メモリの状況で一度テキストをセーブしてエディタのメモリを解放しコンパイラを呼び出す，直接オンメモリで処理する，の両方を使い分けるといったことも実現されるべきものです。

ウィンドウ間のデータ通信（おそらくはパケット用のメモリハンドルのやり取りを中心とした）はSX-WINDOWの中心的な機能（になってほしいと個人的に思っている）です。メーカーのドキュメントを読むかぎりでは，タスク間通信に消極的な位置づけしか与えていないようです。しかし，その可能性たるやはかりしれないものがあります。マネージャと基本フォーマットの確立が望まれます。

## 問題は？

さしあたっての障害はメモリ容量の問題と画面解像度の問題です。

メモリは増設すればすむ，というか増設するしかないので，あきらめましょう。アプリケーション側でメモリ常駐部は基本ドライバのみとし，メイン処理部や追加機能は必要時だけ呼び出すようにして役目を終えると「メモリを解放してもいいよ」指定にしておくことも考えられます。しかし，ちゃんとしたアプリケーションがあれば4Mバイトが「ちょっと狭い部屋」に感じられるのは時間の問題です。

次に解像度。今回の画面写真はほとんど高解像度(1024×848ドット，24kHzインタレース)で撮影されています。通常の768×512ドットも世間では高解像度のうちなのですが，ウィンドウ環境で使うには狭苦しさを感じます。

幸い（？）にもX68000の標準ディスプレイの多くが24kHzに対応していますので，

　A>SXWIN －G18

のようにすることで高解像度の立ち上げも不可能ではありません（使用に耐えるものではないのですが)。長残光ディスプレイを使用することで，実用レベルまでチラつきを抑えることもできます。ウィンドウ環境としてはこれくらいの画素数が必要ではないでしょうか。

これくらいの解像度になるとグラフィックも16色あればたいていのことはできるようになります(カラーページ参照)。バッファ内だけ65536色にしておけばレイトレだってできるでしょう。

## 信頼性

マイクロソフトとIBMでは，デモのたびに互いにWINDOWSとOS/2でどちらがいかに簡単に暴走するかをアピールするといいます。要するにどちらもそんなに安定していないわけです。また，昔からMacintoshがよく飛ぶというのはユーザーの常識となっています。

GUIだなんだとやっているとOSは複雑な動作を要求されることもあるでしょうし，マルチタスクなどともなると，これはもうどこでどんな間違いが起こるかもしれません。

昔は32ビット機クラスになるとまともなOSが備わっているのが当たり前だったのですが，パソコンレベルではまだまだセキュリティへの関心は低いようです。なかにはCPUレベルでせっかくメモリ保護機構を持っていてもみんなスーパーバイザで動作させるOSまであります。

マルチウィンドウやマルチタスクを売り物にするには，システムの信頼性を高めなければなりません。並列で実行されているタスクのうち99個が完全にデバッグされた無害なプログラムでも，1個の動作不安定なプログラムのために復帰不可能な暴走状態を招く可能性もあります。

プログラム自体に信頼性が求められるのは当たり前です。しかし，絶対的な安全はありえないかもしれません。ある程度の大きさになるとバグのないプログラムを作ることは非常に困難なことを私たちは知っています。

そこで重視したいのは「データ保護」に対する考え方です。SX-WINDOWの最大の欠点は使用中にシステムエラーが発生するとリセットを強制される点でしょう。バスエラーを起こしたタスクだけ殺すのではなく，SXシェルを抜けるのでもなくリセットをしなければなりません。「こまめにセーブ」というのはMacintoshの作法（？）ですが，そんなところまで倣いたくはありません。

おそらくメモリ上にそのまま残っているであろうデータを見殺しにせざるをえないのは心が痛みます。しかし，どのアプリケーションがどの領域をどう使用していたかなどはリセット後には知りようがありません。しかし，システムが動いていたときはどうでしょう。実はメモリマンは知っていたはずなのです。それを記録しておけば暴走後でもメモリ内のデータを追うことができるかもしれません。これは十分に可能なことに思われます。「データの回収」のための機能は広い目で見ればシステムの信頼性にもつながってくるはずです。そういった機能があってこそ，高度な機能も生きてくるのではないでしょうか。

なにが必要なのか?

# SX-WINDOWの可能性

Ogikubo Kei 荻窪 圭

SXシステムが当たり前のウィンドウシステムとして成熟するためにはなにが必要でしょうか。ここではシステム的な面からSX-WINDOWに必要なものをまとめてみましょう。新しいコンピュータ環境は見えてくるでしょうか。

「あなたはコンピュータになにを求めるか」,という問いは根本的なところで間違っている。我々はなにかを求めてコンピュータを作ったわけではないのである。強いていえば,弾道計算だ。そうじゃなくて,コンピュータができてみて初めて,その可能性に恐怖し,あれもできるこれもできると用途が膨らんでいったのである。

いまでも状況は変わらない。なにか初めから目的があって,そのためにコンピュータを必要とするのだ,と信じる人の手によって,コンピュータは退屈なものにさせられそうだ。そういう人たちとて,ワープロやらスプレッドシートやらという形を見せられて初めて,コンピュータになにができるか知っただけなのである。人の想像力はそれほど優れたものではない。

まず,コンピュータがあって,OSがあって,そこで初めて人間は考えたりひらめいたりするのである。そもそも,コンピュータというのはそういう不思議な機械なのだ。

コンピュータに新しいプラットフォームが乗る,というのは,新たなる可能性を見いだせ,ということである。いままであったコンピュータの仕事がプラットフォームが変わることによって便利になる,といった考えではなく,新たなプラットフォームができることによって,また新たな概念が生じると思わねばならない。そうでなければ,プラットフォームが変わる意味がない。

## 前編:SX-WINDOWはどうなっていくべきか

SX-WINDOWというものは登場したが,アプリケーションは情けないEasyPaintだけだし,速くはなったが,ついてくるアクセサリもエディタも完成度は高くない。以後登場するべきアプリケーションの礎になれるだけの思想も足りない。

もっとも,いまの状況で本格的なアプリケーションなんて登場した日には,目も当てられない。ほとんどのユーザーがメモリ不足で,非常にシビアな状況でそのアプリケーションを開くことになるだろう。

しかし,ある意味でいい状況でもある。アプリケーションの肥大を防げるからである。小さなアプリケーションを組み合わせてなにかを作り上げるという,Macintoshや WINDOWS 3.0が忘れ去ってしまった世界が可能だからだ。SX-WINDOW上のアプリケーションはすべてが道具であり,部品である。ユーザーは好きな部品を取り出して環境を作り,好きな道具を組み合わせて目的を達成する。すべてはそこから始まるのである。

それには多くの共通フォーマットが用意されなければならない。

本来なら,ウィンドウシステム上にその人が使いたいものが常に広がっているのが理想である。他のウィンドウシステムでは,メモリの問題や画面解像度の問題でそうはいかない。Macintoshを使っていると,目的のデータを見るために,メモリを圧迫している他のアプリケーションを閉じてから,そのデータを開くという面倒な作業が必要になる。WINDOWS 3.0の狭い画面にいたってはなにをかいわんや。そもそもWINDOWS 3.0のウィンドウの設計自体が,画面を狭くするようになっている。

だが,SX-WINDOW自体,完成されたものではない。まだ,新しいプラットフォームとなるには不完全な部分が多く存在する。

やることはたくさんあるのである。シャープだけで開発するのが大変ならば,アップル社のやることを見習えばいい。たとえば,MODE32というソフトがある。これはROMが古いマシンでもSystem7.0の仮想記憶を使えるようにするソフトである。サードパーティが開発したそれをアップルは買い取って,ユーザー全員にタダで配ったのだ。さらに,ATMというアウトラインフォントを使うプログラムもアドビと契約して,新しいシステムにタダで添付することにした。このように,フリーウェアだろうがなんだろうが,ユーザーにとって必要だと思われるものが登場したら,シャープは買い取って,ユーザーにタダで配ったり,次のシステムへ添付するというのはひとつの手である。

## データの使い回し

誰がなんといおうと,各種データを統合された環境で扱うことは必要である。各々の形式のデータに標準フォーマットを設け,どれもクリップボードで扱えるようにする。いまはテキストデータだけで,それではいままでと変わらないではないか。

1) テキスト

テキストだけでなく,文字の属性,修飾なども標準化したい。こういうのをリッチテキストフォーマットと呼ぶらしいので,SX-WINDOW版リッチテキストフォーマットがあるといいだろう。

2) ビットマップデータ

PIXデータがあるにはあるが,圧縮されたフォーマットがほしいところだ。さらに,ビットマップデータもクリップボードに入れ,なおかつひとつのグラフィックが文字と同様に扱えるようになるべきである。

3) ドローイングデータ

こちらも重要である。ビットマップデータより重要なくらいだ。

4) PCMデータ

PCMデータもクリップボードで扱えるべきだし,ワープロやエディタ中にペーストできるべきである。たとえば,PCMを示すアイコンが挿入され,クリックすると音が出る,という感じだ。

5) OPMデータ

OPMデータも同様。使い回せる形を整えるべきである。クリップボードへ張り付けるのだ。

6) ボタン

発展形として,ボタンデータも考えてみ

た。ボタンは1)〜5)のどんなデータでも混在できるほか，バッチファイルのような扱いもできる。ボタンにはプログラムや一連のマクロを割り当てることができるのだ。

## マルチフォント化

12ドットとか16ドットというのはコンピュータ側の都合である。ユーザーにとって必要なのは，画面上のフォントだけでなく，印字されたときのフォントだ。そうなると，アウトラインフォントが重要になってくる。書体倶楽部でもいいが，あれはちょっと質がいまひとつなので，なんとか，日本語ATMだとか日本語TrueTypeといった他社の技術を流用するのがいいだろう。フォントが自由になると（それが明朝とゴシックだけでも）世界が変わる。フォントやその大きさはコンピュータの都合ではなく，人間の都合で変更できるようになるべきだ。

少なくとも，このレベルになるとシステムがサポートしてくれないとどうしようもない。テキストマネージャやフォントマネージャという問題になってくるからだ。

## デスクトップにアイコンを置けること

アイコンはウィンドウの中だけにある，というのは間違いである。よく使うアイコンは，デスクトップ，つまり，本棚や引き出しではなく，机の上に置いておきたいものである。これはMacintoshの知恵だ。

## エイリアス機能

UNIXのエイリアスとはちょっと違う。ここで考えたのは，MacintoshのSystem 7.0で採用されたエイリアスだ。Macintoshでは，ファイルなりフォルダなりのアイコンを選択して，メニューからmake aliasを実行すると，名前が斜体になったダミーのアイコンができる。これが，エイリアスアイコンである。こいつがあると，アプリケーションがどこにあっても，また，ファイルがどこにあっても，ダブルクリックすればちゃんと本体にアクセスしてくれる。すると，データを入れるディレクトリすべてにエイリアス化されたアプリケーションのアイコンを置いておくと，便利である。ディレクトリのエイリアスアイコンは，デスクトップに置いておく。すると，どんなに深い階層のディレクトリでも，デスクトップから呼び出せる。

ファイルの管理がいちだんと楽になるのである。WINDOWS 3.0のプログラムマネージャなんて目じゃないのである。

さらに，バッチファイルのように，複数のアプリケーションをパイプでつないだものをまとめてエイリアス化できたりすると，もっといいかもしれない。

## 仮想記憶のサポート

SX-WINDOWは，たいていの人がそうしているとおり，アプリケーションを開いたり閉じたりはあまりしない。必要なものは，デフォルトで開きっぱなしである。と，なると，大きなアプリケーションが増えてきたとき，メモリが足りなくなるのは明白である。こういうとき仮想記憶は役に立つ。もっとも，ある程度のメモリは積んでいないと，がしがしとハードディスクが回りっぱなしでうっとうしいだけであるから，実メモリは4〜8Mバイトくらい積んでおきたいけど。

## エンドユーザーレベルの開発環境

WINDOWS 3.0にはVisual Basicというものが登場した。WINDOWS 3.0上で動くBASICである。WINDOWS 3.0特有のユーザーインタフェイスはマウスを使って簡単に構築でき，あとは処理を記述するだけというBASICだ。SX-WINDOWに真っ先にほしいのはそういったエンドユーザーレベルの開発環境である。当然，コンパイルできるべきであるが，BASICインタプリタが常駐していれば，自動的にSX-BASICのアイコンを判断して，実行してくれるという形でも状況は随分変わるだろう。

## 後編：まったく新しいシステムへの脱皮

たとえば，ビル・アトキンソンの考えるシステムには，アプリケーションは存在しないらしい。あるのは作業する場と，道具だけである。従来アプリケーションと呼ばれていたものには道具が該当する。つまり，紙に対して，任意の道具を引っ張ってくれば，その紙は計算用紙にもスケッチブックにも電話帳にもエディタにもなんにでもなるのである。

これまでのウィンドウシステムではアプリケーションが紙と道具の両方を用意していた。すると，アプリケーションごとに異なった紙を使うので，その融合が極めて困難になるうえに，資源の無駄遣いとなる。

SONYのパームトップはなかなか頑張っている。3種類の紙と，いくつかの道具しか与えられない閉じたシステムではあるが，紙が1種類になり道具がプラグイン式にあとから追加できるようになれば，立派に新しいシステムができあがる。

ビル・アトキンソンの考えるシステムはちゃんと提示されたものではないので，私が多くを想像で補完してしゃべっているのであるが，たとえば，紙があるとしよう。

紙を管理するシステム（目次や索引）がある。紙にかぶせるテンプレートがある（世間の紙にも，方眼紙や罫入りの用紙や領収書や財務諸表などいろいろあるではないか）。店に行くと，あらかじめいろいろな情報が吹き付けられた紙を売っている（本のメタファーだ）。

そして，紙に対して情報を書きつける道具がある。この道具が紙と鉛筆のメタファーを超えるさまざまな可能性を秘めているのである。その道具とは，文字を書くためのペンであり（キーボードも手書き入力も可能だ），絵を描くためのセットであり（ペン画もあれば，油絵もあるし，レイトレもある），音を入れるための録音再生セットであり（ここでいう紙にはなんでも入るのである），動画セットであり，計算機であり，コンパイラであり，スケジューラであり，通信装置である。もちろん，ユーザーが自分で道具を作るための道具もある。MIDIデータを記述する道具もある。紙をゲームボードにする道具ももちろんある。紙は共通である。ときには，紙の上の情報がビデオなりMIDI音源なりに転写される。テレビにもなる。DTPもDTMもDTVもDTAもなんでもこいだ（DTAってのは，デスクトップアミューズメントらしい）。

さらに進むと，紙の上に書きつけられたモノすべてがオブジェクトになり，ほかの紙の上にあるオブジェクトとリンクさせたり，紙同士の関係が柔軟に記述できたりしてハイパーな構造が作られる。オブジェクトはダイナミックデータリンク機能（あるいは，ライブコピー&ペースト）により，ひとつのオブジェクトがいろいろな紙に存在することが許される。

ユーザーは目次や索引から紙を選び，参照したり，加工したり，新しいページを作ったりする。

話は飛んできたが，これは私の考えるDynabookでもある。SX-WINDOWにはWINDOWS 3.0のように自分のウィンドウの中に自分の紙を広げるようなシステムではなく，こうなってほしいものである。

X68000の場合=これまでとこれからを考える

# SX-WINDOWを検証する

Kioi Makoto 紀尾井　誠

まだまだ，海のものとも山のものともつかないSX-WINDOWですが，可能性は大きなものを秘めています。ここでは現在のシステムの改善すべき点，改良できる点を挙げて「ちゃんとした」ウィンドウシステムのあり方を考えてみましょう。

SX-WINDOWが私たちの手元にやってきてからすでに2年近くがたとうとしている。その間に起こった出来事はver1.10へのバージョンアップとEasyPaint SX-68Kの発売のみ……。ちょっと少なすぎるといっていい。ここでもう一度SX-WINDOWの課題を検討してみたい。

## データフォーマットの標準化

グラフィックに関してはSX-WINDOW登場のときから何度もパレットの問題が指摘されている。

はっきりいうと，各ウィンドウが16色の任意パレットを使うということは「画面全体でひとつのグラフィックしか表示しない」という意思表示と同義である。同時にひとつのグラフィックウィンドウだけでいいのなら，65536色のものを1ウィンドウだけ使うほうがはるかにX68000らしい。65536色はX68000の大きなアドバンテージなのだから。そうなればウィンドウ上のアプリケーションもはるかに楽しくなる。カット&ペーストで色変換などの心配もなくなるし……。

しかし，統一パレットで作業することを中心としない限りマルチウィンドウ環境が生きてこないのは明白だ。対応するグラフィックツールがあればよいのだが，EasyPaintはそのあたりについてはほとんど考慮されていない。もっとも，SX-WINDOW以外のウィンドウシステムでは解決されている問題かというとそうでもない。

いずれにせよ，さしあたって必要なのはグラフィック画面ではなくテキスト画面用(モノトーン4階調)のグラフィックエディタだ。万一，データフォーマットの統一が取れなくても，使い回しはきくはずだ。モノトーンとはいえ4階調の表現力も侮れないし，データの利用価値ははるかに高いと思われる。

データの標準化という作業には，2つのアプローチがある。データフォーマットを本当にひとつにする方法と，さまざまなフォーマット間で相互変換する方法だ。後者はアプリケーション側に負担がかかるがユーザーには関係ない。

あまり練り込まれてもいないものを標準だよといって提示されるより，野放しのほうがマシかもしれない。標準は「とりあえずの標準」にすぎない。強いものが勝つ。よりよいものが残る。それが当たり前。Macintoshの世界ではアップル社より強い奴がほとんどいなかっただけの話だ。

「Macintoshではアプリケーションが違っても操作性が統一されている」というのも神話にすぎない。圧倒的に強いものがあれば残りがそれに倣うだけだ。ちなみに，そのように従順なプログラムは「Macintoshらしいプログラム」と呼ばれる。

## 画面構成

NeXTと似ているためかSX-WINDOWの画面まわりを悪くいう人はほとんどいない。グレイ（もしくはモノトーン）4階調で作られた画面をセンスのよい画面だと勘違いしている人も少なくない。確かにMS-WINDOWSなどよりは上品だが，モノトーンの使い方がいまいち下手なのだ。

全体に暗く，ウィンドウとデスクトップの区別がつきにくい単調な画面構成……。なにがマズイかを突き詰めれば，地色を薄いグレイとしていることに帰着する。実際に使用してみると「背景色がグレイ」というのはあまりいいものではない。画面全体のコントラストが足らないので文字が読みづらくなり，結果的にディスプレイの輝度を上げて目の負担を増やすことにもなりかねない。網掛けされたOh!Xのリストページも実は評判がよくない。

また，ハードコピーを取ったときに汚くなることもあまりうれしくない。WYSIWYGだなんだという時代だ。出力のことまで考えてみれば，ウィンドウ内は白地しかありえない（ちゃんとした網掛けを出力するには1000dpiくらいは必要だ）。

もっとも，これは作成されるアプリケーションが個別に対応すれば解決する問題ではある。じゃあなぜ，みんな白地にしないかといえば，薄いグレイならば表示が簡単になるようにシステムが構成されているからだ。うーむ。

## ハードウェアの使用状況

SX-WINDOWではX68000のハードウェアが十分に生かされているわけではない。いくつかの機能を殺してウィンドウ環境を実現しているのだ。どれだけの代価を払っているのか，どれくらい回収できるのかを見てみよう。

まず，スプライト。SX-WINDOWでは通常横768ドットモードが使用されるから，ハードウェア的な制限からスプライトは使用できなくなる。以前，SXGMODE.Xで，スプライトの使用もまるっきり不可能ではないということは示されたが，マルチタスク環境での管理となるとまた話は変わってくる。

スプライトが使用できるように画面モードを変更したとしても，ほかのアプリケーションで同じスプライトを使用していたら表示はとっても変になる。対策には，

1) アクティブウィンドウのみ使用を許可
2) マネージャを作って管理

の2通りの方法が考えられる。マネージャがあっても悪くないと思うが……。

そのほかにもFM音源なども多数のタスクで使うわけにはいかない資源だ。プリンタだってちゃんとスプールしなければならない。マルチタスクっていうのはかなりうっとうしいこともたくさんやらなければならない。ちゃんとしたシステムならそのあたりのところも整備しておかなければいけ

ないだろう。

見た目の問題とも関連するのだが，ここで「THE WINDOWS」創刊号のMS-WINDOWSのGUIに関する大島篤氏の文章を引用しよう。

「ウィンドウどうしが重なったときに，立体感が十分でないように思うのだ。ペラペラの紙にだって厚みがあり重ねれば紙のエッジが光ったり影ができて立体感がわかる」

DOSマシンに要求するには，なにか無茶な言い分のようにも思える。しかし，前述のように同じことをSX-WINDOWを使っていて思っていたのも事実。同じようなウィンドウがたくさん開いていても，乱雑にしか感じられない。

いろいろ考えた結果，SX-WINDOWの場合はまだ改善する余地があると思う。たとえば，ウィンドウは4階調のモノトーンで描画されている。ご存じのように，これはコントロールパネルで色調を変更できる。もし，アクティブなウィンドウだけ違う色調の4階調だったら，かなりわかりやすくなるのではないだろうか。

SX-WINDOWでのテキストVRAMの使用状況を図に示す。このうち2～7の色はSX-WINDOW上では表示されることはない（ドキュメントでは8以降を使えとある）。ちょっともったいない。やりようによっては，デスクトップ（背景），ウィンドウ，アクティブウィンドウをそれぞれ別系統の色でまとめることもできそうだ。特に不都合はなさそうなのでアプリケーション側で無理矢理対応ということも……今後の課題かもしれない。

## ユーザーインタフェイス

いまだにMacintoshのユーザーインタフェイスを最高のものと信じている人もいるようだが，なんといってももう8年も前に作られた仕様だ。すでに古さは隠しきれない。それを超えるものがなかったという事実のほうが情けないくらいだ。

SX-WINDOWには「クリップボード」というものが設定されている。これを使えば別のアプリケーションでちょこちょこと作った図などが簡単に読み込める（ということになっている）。Macintosh神話を作った偉大な機能のひとつだ。シングルウィンドウ，シングルタスクがMacintoshの基本だったので，アプリケーション間でデータをやり取りする際にはどうしてもワンクッション置く必要があったのだろう。マルチウィンドウ・マルチタスクのSX-WINDOWの場合はどうだろうか？　現状ではそういうアプリケーションもないのだが，基本作法から推察すれば，データをひっつかんでそのまま放り込めばいいことになる。クリップボードの出番がくるのはメモリが足りなくて2つのアプリケーションが同時に開けない場合だけだが，そういう環境はふつうマルチウィンドウとは呼ばれないだろう。SX-WINDOWは本来クリップボードなどという面倒なものは必要ないシステムのはずなのだ。

まだ不満が残る点もあるが，ファイル操作関係でのSXシェルの機能はトップレベルにあるといっていい。しかし，システムリソースの拡張という大袈裟な方法でしかカスタマイズできないSXシェルは昔のビジュアルシェル（vs.x）に大きく劣るところがある。ウィンドウの位置やアイコンの配置を覚えているというのも一長一短。

機能を見れば，SXシェルはビジュアルシェルの欠点を改良している。しかし，ビジュアルシェルの美点についてはあまり学んでいないように思われる。

＊　＊　＊

以前Oh!Xで「SX-WINDOW環境セットアップ」という記事が予告されたことがあった。しかし，この予告は実行されなかった。どうも，あまりにもやることがなかったらしいのだ。SX-WINDOWをインストールするにあたっては，フロッピーだろうがハードディスクだろうが，ほとんどなんの手間も必要としない。FSX.Xを組み込み，SXWIN.Xを起動するだけ。それ以上でもそれ以下でもない。操作にあたっても，当たり前のことをするだけ。難しい部分や不自然な部分はほとんどない。

MacintoshやMS-WINDOWSをハードディスクにインストールするにはそれなりの知識やよくできたインストーラの力が不可欠だ。SX-WINDOWのこういう部分はものすごくよくできている。

## ユーザーインタフェイスガイドライン

ユーザーインタフェイスガイドラインというものがある。どういう操作性のソフトを作ればいいかをまとめてある本で，GUIを持つOSには必ずついてまわる。Macintosh, MOTIF GUI, OpenLookGUI……。書店で入手できるものだけでもかなりの種類があるが，細々しすぎていて読んでもあまり面白いモノではない。ただ，MacintoshのHyperCard用スタックのユーザーインタフェイスガイドラインはエンドユーザー向けに書かれており，内容も面白いのでとりあえずおすすめしておく。

SX-WINDOWにはまだこのようなものはない。貴重な解説書「SX-WINDOWプログラミング」で触れられているガイドラインは著者がシャープに問い合わせて書かれたものなので現在のところ唯一の手がかりといえる。SX-WINDOWのドキュメントは一応の目安にはなるものの，標準システムのアプリケーションを見ても全然参考にならないのでいまひとつ信頼性に欠ける。

＊　＊　＊

SX-WINDOWというのは結構ちゃんとできたシステムだ。基本的な考え方に慣れれば従来のシステム上のアプリケーションをそのまま載せ換えることはそれほど難しくないだろう。現実問題としてSX-WINDOWへの移行がなかなか進行していないのは2つの原因によると思われる。ひとつはメモリの問題。そして，次に，どんなアプリケーションを動かすべきなのかというビジョンが与えられていないこと。

どんなかたちのものであれ，システムは必ず思想を伴って現れる。人を動かすのはシステムの機能や性能ではなく思想の部分である。無論，メーカーのビジョンがなくてもアプリケーションを作ることは可能だ。SX-WINDOWには魅力あるアプリケーションが揃うべきであるし，いろいろなものが達成できると思う。が，これだけ大掛かりなものを作ったのだから，もっとちゃんとしたかたちでシステム，開発環境，ガイドラインが提供されることを誰もが待っている。それがSX-WINDOWの不完全さを際立たせているといえる。実際，現在のバージョンはまだ評価版程度のものとしか受け止めていない人がほとんどだろう。

SX-WINDOWはまだまだ真の姿を見せていないのだ。

| パレット番号 | 色 |
|---|---|
| 0 | 透明 |
| 1 | 明るいグレー |
| 2 | 暗いグレー |
| 3 | 黒 |
| 4 | 黄色 |
| 5 | 赤 |
| 6 | 緑 |
| 7 | 青 |
| 8 | 白 |
| 9 | 明るいグレー |
| 10 | 暗いグレー |
| 11 | 黒 |
| 12 | 黄色 |
| 13 | 赤 |
| 14 | 緑 |
| 15 | 青 |

オペレーションとデータの統一

# Macintosh OSに学ぶ

Ogikubo Kei 荻窪 圭

ウィンドウシステムの現在・過去・未来のうち，現在ではMacintoshが最高の評価を得ています。Macintoshのシステムがsx-WINDOWに与えた影響もはかりしれません。ここでMacintoshの成し遂げたことをまとめてみましょう。

Macintosh ClassicとMacintosh LCのおかげで，Macintoshユーザーは着実に増えつつある。ユーザーすべてがMacintoshをMacintoshらしく使っているかどうかは別にして，である。もしかしたら，一緒に購入したワープロしか使っていないかもしれないし，有効に利用するオペレーションを発見することができず，「Macintoshって面倒くさい」と思っているかもしれない。

さて，そんなMacintoshであるが，初代Macintosh（なんと，メモリが128Kバイトしかなかったのにウィンドウシステムを動かしていたのだ！）から早7年になる。その間，システムはどんどんバージョンアップを重ね，とうとうバージョン7.0まで到達した。が，今回解説するのは，バージョン7（System7）ではなく，その前の，System 6.0.7である。理由は明白だ。まだ日本語版（漢字Talk 7.0）が完成していないからである。System7.0の話もまじえるが，原則的には6.0.7だ。

## Macintoshのシステム構成

Macintoshはツールボックスやクイックドローなどが入ったROMとディスクから読み込むシステムによって動かされる。

ROMには64K，128K，256K，512K，1Mバイトの5種類があるが，新しいROMでサポートされた機能はディスクによって旧ユーザーにも供給されるので問題はない。

システムはハードウェアインタフェイスドライバ，リソースマネージャ，クイックドロー，ユーザーインタフェイスツールなどによって構成される。ユーザーインタフェイスツールにはSX-WINDOWのアイデアの元になったダイアログマネージャやイベントマネージャ，テキストマネージャといったマネージャが幅をきかせている。

細かい話は（私もよく知らないので）省くとして，重要なのは，これらマネージャをフルに活用してプログラムを構築していかなければならないことである。ハードウェアを直接叩くなどということは絶対に許されないし，そんなことをしたらMacintoshが無節操な新機種を出すたびにプログラムの書き直しが必要になって，かえって手間である。さらに，ユーザーインタフェイスガイドラインという書物があって，Macintosh用のプログラムを開発する際はこのガイドラインにそったユーザーインタフェイスのプログラムにしなければならない（しなかったからといって警察につかまるわけではないし，100%守っているプログラムばかりではないが，ある程度守らないと，ユーザーに怒られる）。

このあたりがキモである。

## Macintoshのシェル

Macintoshはキーボードによるコマンド入力を一切許さないユーザーインタフェイスをかぶせたOSを持っている。マウスによるオペレーションもできる，というのではなく，マウスによるオペレーションしかできないのである。

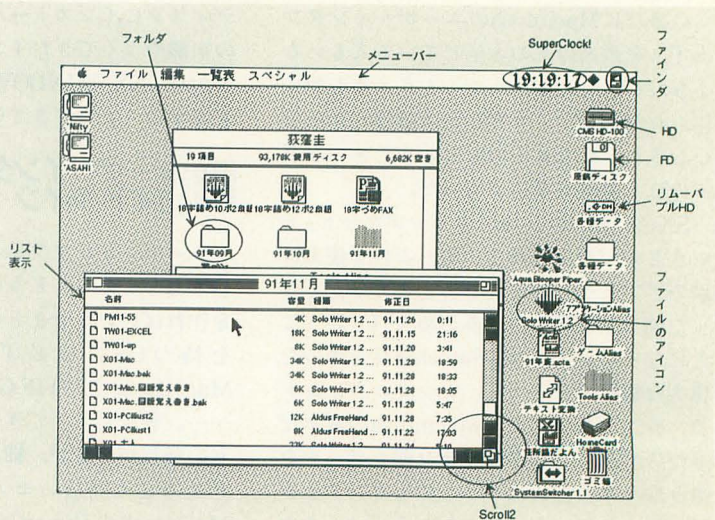
Macintoshのウィンドウシステムというと，メニューバーがあって，ウィンドウがあって，ゴミ箱があって，という画面を思い浮かべるだろうが，これは，OSにのっかっているシェルにすぎない。このシェルはファインダという。アプリケーションが実行されるときは，ファインダをメモリから消して，代わりにアプリケーションがいすわることになっている。

ファインダにはアイコンがつきものである。このアイコンの管理はシステムが行うのである。実際には，アイコンはIDで管理され，すべてのアイコンはDeskTopという不可視ファイルに登録される。ここにアイコンが登録されていれば，ファイルはアイコンの形で表現されるし，そうでなければ（あるいは，アイコンが多すぎて，登録できないこともある），ただの白い紙のアイコンになる。DeskTopは記憶デバイスひとつにつきひとつだけ作られる。DeskTopにはその他，ウィンドウの情報なども格納されるようだ。SX-WINDOWでいうDIRDTOP.SXというファイルと同じと思っていい。

Macintoshのユーザーインタフェイスはデスクトップ，つまり机上のシミュレートで構成されている。

## Macintoshの基本オペレーション

Macintoshのシステムはマウスとキーボードで操作する。ジョイスティックや音声

図1

でコントロールすることは標準的ではない。

マウスはボタンがひとつのワンボタンマウスである。キーボードは（あろうことか，機種によって存在するキーや配列が違うのだ！），原則として，ファンクションキーやINSキーはなく，optionキーやコマンドキー（アップルマークがついている）がある。コンパクトで嬉しい。拡張キーボードというIBM PCを真似た配列のでっかいやつもあるが，こいつはでかいうえにいらないキーも多く，肝心なキーが遠かったりして，私は嫌いである。

で，shiftを押しながらドラッグとか，optionを押しながらプレースなど，shift, option, control, commandキーはマウスと併用して機能の増加に対処している。

メニューバーはプルダウンメニューである。つまり，ボタンを押している間だけメニューが出ている。ドラッグするとマウスポインタのあるところが反転し，指を離すと実行する。使えない機能は，グレイ表示されている。サブメニューがある場合は右向きの三角が，ダイアログが開く場合はコマンド名の後ろにピリオドが3つつく。この表現方法は，いろんなウインドウシステムやアプリケーションが真似している。

アプリケーション内でのドラッグは，そのときに選ばれているツールで意味が変わる。矩形を描くツールが選択されていれば，ポインタが十字になり，ドラッグして矩形を描ける。文書であれば，Iビームポインタという形になり，文章を選択できる。図形をドラッグすれば，そのドラッグ範囲にある図形を選択できる。shiftを押しながらドラッグすると，いくつものオブジェクトを指定できる，という寸法だ。

マウスでオブジェクトを指定すると，次はメニューバーからそのオブジェクトに対する操作を決定する。カットしてもいいし，その範囲のフォントを変更したり，色を変えてもいい。この辺はアプリケーションによって，指定したオブジェクトがアイコンかテキストか図形かによって変わるのだ。

指定したオブジェクトが図形やアイコンであれば，ドラッグして動かすことも可能だ。これでファイルのコピーや移動を行う。同一メディア内ならば，コピーではなく移動だが，同一メディア内で別フォルダ（ディレクトリと読み替えていい）にコピーするときは，optionキーを押しながらドラッグする。

で，慣れてくると，いちいちマウスで全部指定するのが鬱陶しくなってくるのは，誰もが指摘する通りだ。で，Macintoshでは，ショートカットキーを使える。コマンドキー（ キー）＋英字キーというのが基本的な組み合わせだ。アプリケーションによってはshift＋ ＋アルファベットというような複雑な組み合わせを指定できたり，ショートカットキーをカスタマイズできるアプリケーションもある。

よく使うショートカットキーはある程度決まっている。Undoは＋Zで，カットは＋Xで，コピーは＋Cで，ペーストは＋Vという感じだ。この配列は，非常に重要である。コマンドキーはスペースキーの左にある。ZXCVというのは，キーボードの左下に並んでいる。つまり，右手でマウスを持って範囲を指定したり，オブジェクトを選択し，空いている左手でカットしたりコピーしたりができるという無駄のない配置なのだ。少なくともカット＆コピーという頻繁に使うものに対しては，右手と左手を分担させて，素早く処理できるのである。奥が深い。これが，SX-WINDOWのように，ショートカットキーにOPT.1などを使われた日には，鬱陶しいだけで，だったらマウスで右ボタンポップアップメニューのほうが簡単だ。

さらに，テキストの場合，shift＋矢印キーで範囲指定できることになっており，キーボードだけでカット＆ペーストは可能だ。

以上がMacintoshの操作の基本である。わからなければ，クリックしたりダブルクリックしたり，option＋クリックしたりといろいろとやってみる。やばい，と思ったらすぐにアンドゥする。という原則で，好奇心とちょっとばかりの想像力と思考能力があれば，たいていのソフトは勘と試行錯誤で扱えるのだ。

## Macintoshのファイル・ディスク管理

Macintoshも見た目は派手でも，内部ではDOSやHuman 68kのように，あるいはそれ以上地味な処理を行っている。ファイルや記憶デバイスの管理もそうだ。

Macintoshではすべてのファイルは2つのフォークに分かれている。リソースフォークとデータフォークである。リソースフォークしかないファイルやデータフォークしかないファイルもあるが，とにかく，ひとつのファイルはリソースフォークとデータフォークに分かれているのだ。

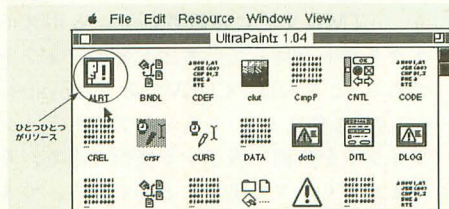



図2

リソースフォークはリソースが記述された部分である。リソースというのはアイコンの形であり，ダイアログやメッセージであり，（原則として）プログラムそのものだ。Macintoshのプログラムはコンパイルすると，リソースになる。

Macintoshのプログラムはすべてリソースフォークを持っている。リソースフォークだけのものもある。リソースとはなにか。説明は困難なのだが，訳すと，資源だ。共立出版のOSシリーズの6巻，「Macintosh」にはこう書いてある。“いわば，Macintoshが標準と定めたデータの構造体と考えることができる”。わかる人はわかるだろう。

リソースフォークには複数のリソースが存在しているのが普通だ。アイコンのリソースやダイアログのリソース，ウィンドウの定義のリソースなどシステムが定めたものもあるし，開発者がそれ以外のリソースタイプを定義してもいい。たとえば，とあるアプリケーションのリソースフォークは図2のようになっている。これは，ResEditというリソースエディタ（アップルが配っている）中の図である。

対して，データファイルの場合，アプリケーション独自のデータ形式で，データフォークしか使っていない場合が多い。ただし，データファイルがリソースフォークを持っていてもよく，SoloWriterの文書ファイルは，リソースフォークに属性や文書情報，フォント情報などを蓄え，データフォークにはテキストファイルだけと分散させている。さらに，TYPEがTEXTなので，SoloWriterで作ったファイルはネイティブのままでテキストファイルとしても使えるのだ。これはアイデアの勝利である。

### デスクアクセサリのこと

メニューバーの一番左にあるアップルメニューは，DA（デスクアクセサリ）用のメニューである。マルチファインダでなかった頃，いつでもどこでも使えるちょっとしたツールを格納する場所として用意された。DAはすべてDAの形式のプログラムであり，FONT/DA MOVERでシステムに組み込まれる。

コントロールパネルやセレクタ（プリンタなどを選択する），スクラップブックのほか，サードパーティ製品やフリーウェア/シェアウェアには，文字コード表，グラフィックツール，エディタ，ファイル管理ツール，単位コンバータ，ゲーム，英和辞典（rSTONEという製品があり，私はいつもお世話になっている）など実に多彩なプログラムが揃っている。

## Mac OSはマルチタスクか

Macintoshはウィンドウシステムではあるが，元々マルチタスクなどではなかった。ファインダから起動したプログラムはファインダの代わりにメモリを支配し，終了するとまたファインダが呼び出されるというシングルタスクのシステムだったのだ。

System 6.0.4からマルチファインダという新しい仕組みが使えるようになった。ファインダをメモリ上に常駐させたまま，他のアプリケーションを立ち上げられるシステムだ。後ろのウィンドウやデスクトップが透けて見えるという，いわゆるウィンドウシステムといって想像できるようなもの。図1はマルチファインダで立ち上げてある。右上の，現在アクティブになっているプログラムを示す小アイコンが証拠だ。

Macintoshの場合，アプリケーションによって画面最上位のメニューバーが支配される。それはマルチファインダになっても変わりなく，メニューバーの様子は，一番左のアップルマーク以外はそのときアクティブになっているアプリケーションのものだ。Windows 3.0やSX-WINDOWは，アプリケーションは自分のウィンドウを持つ。SX-WINDOWに至っては，面倒なことに，アプリケーションが複数のウィンドウを持つことさえ当たり前だ。Windows 3.0では，アプリケーションのウィンドウの中にデータファイルのウィンドウが開くという入れ子になっていて，ただでさえ狭い画面を無駄に使おうとしている。

Macintoshはもっと原始的なので，アプリケーションがメニューバーを支配するのだ。マルチファインダでは疑似マルチタスクをサポートしているので，非アクティブなアプリケーションのウィンドウも裏で動くことは可能だが，原則として，その画面をアクティブなアプリケーションが支配する。また，同じアプリケーションをいくつも開くことはできない。

立ち上がっているアプリケーションの切り替えは，右上のアプリケーションの小アイコンをクリックして切り替えるか，アップルメニューをプルダウンして，そのとき開いているアプリケーションのリストから切り替えたいアプリケーションを選ぶか，ウィンドウの向こうに隠れている別のアプリケーションのウィンドウをクリックしてアクティブにするか，どれかで行う。System 7.0では，右上の小アイコンでプルダウンすると，リストが表示される。立ち上がっている特定のアプリケーションをHideする（見えないように隠す）こともできる。

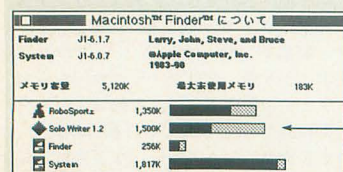



図3

システムがでかいことと（たくさんINIT/CDEVを組み込んでいるせい）マルチファインダしているとメモリがすぐ足りなくなるところに注目

複数のアプリケーションが立ち上がっているときのメモリ管理だが，図3のようになっている。全体のメモリのうち，アプリケーションごとにメモリを確保して立ち上がる。当然，同時に立ち上げられるアプリケーションの数は，搭載しているメモリの量や，アプリケーションの大きさに左右される。仮想メモリがサポートされるのはSystem 7.0からだ。

## Macintoshの日本語表示システム

Macintoshで日本語を扱うには，漢字talkというCDEVを組み込む。IBM PCで日本語を扱うためには(つまり，DOS/Vを動かすには)，日本語を扱うためのデバイスドライバを組み込む，というのと同じである。

さて，Macintoshの持っているスクリーンフォントには2つある。ビットマップフォントとポストスクリプトフォントである。ビットマップフォントには地名がついているので判断できる。ポストスクリプトフォントもスクリーン上と普通のプリンタではビットマップフォントだが，ポストスクリプトのフォント名になっており，ポストスクリプトプリンタをつなぐと，綺麗なアウトラインフォントで印字される。日本語では，ビットマップフォントがKyotoとOsakaであり，プリンタフォントが細明朝体と中ゴシック体だ。

欧米では，True Typeと，ATM(アドビタイプマネージャ）という2つのアウトラインフォント技術をアップルはサポートしている（後者に関しては，正式にサポートすることになった)。TrueTypeはSystem 7.0で標準でサポートされるフォントであり，ATMはすでに市販されているポストスクリプトを持っているアドビのフォントである（こちらは，CDEVであり，System 6.0.7でも使える)。どちらもアウトラインフォントを画面に表示する技術であり，そのままプリンタに出すこともできる。つまり，ポストスクリプトプリンタを使わなくても，アウトラインフォントの綺麗な印字が可能なのだ。日本語版はまだ制作中で，日本語ATMがそろそろ登場しそうである。期待するところ大だ。日本語ATMが登場すれば，安いプリンタでも綺麗な印字が(フォントの展開をMacintosh側で行うので印字速度は遅くなるが）可能になる。

漢字talkの完成度はMacintoshらしくない点を多く抱えている。たとえば，アプリケーションと日本語FEPのインタフェイスが決まっていないこと。だから，アプリケーションが日本語FEPでインライン変換しようと思うと，それぞれのFEPごとにプログラムを書かねばならない。そうでない場合には，変換ウィンドウが開くことになる。

続いて，コマンドキー＋スペースキーで日本語入力モードと英文モードを切り替えるのだが，どちらのモードか知るためのマークが右上の●なのだ。これが●だと日本語入力モードで，◆だと英字モードなのである。見てわかるのが身上のMacintoshなのに，こんな記号で済ませるというのは怠慢である。私はそれがわからなくて悩んだぞ。日本側のセンスの欠如というか，志の低さを物語っているといえよう。System7用の漢字talk7.0ではもっとスマートなアイコンになるはずである。

さらに，訳の統一を図らなかったのも日本側の怠慢（もっとも，アップルコンピュータジャパンが日本でちゃんと頑張り始めたのは最近の話で，当時はキヤノン販売が全部やっていたのだが）だろう。

## Macintoshのデータ共有実態

Macintoshが優れているのは，さまざまなデータの標準的なフォーマットが存在することである。テキストデータはMS-DOSやHuman 68kと同じ程度のもので，テキストの並びをサポートするだけである。アップルはRTF（リッチテキストフォーマット）を作ろうとしているようだ。これは，テキストだけでなく，フォントなどの属性も共通化しようということ。つまり，いまは共通化されていないのだ。困ったものである。

グラフィックデータはそうではない。PICTというモノクロのドローイングデータやイメージデータを扱うフォーマットと，PICT2いうPICTのカラー版がある。さらに，マックペイントというペイントソフトのデータもほとんどのソフトで利用可能だ。ドローイングデータが共通化されているの

である。

Macintoshのワープロのほとんどが，文字の代わりに，PICTファイルをペーストすることができる。上のほうの世界では，ESPフォーマットというポストスクリプトファイルも共通のデータとして扱える。

音に関しては，サウンドリソースというリソースが一応共通形式といえるだろう。

また，多くのソフトが他のファイルのネイティブなデータ形式をサポートしている。ものによっては読み込みだけでなく，書き出しもサポートしている（でも，英語での話である。日本語ワープロに関しては，まだ遅れているのが実情)。

データの使い回しがMacintoshの身上のひとつだ。System 7ではライブコピー&ペーストという一歩進んだ形でのデータ共有を図っている。

## アプリケーション管理の上での欠点

Macintoshのシステムは非常に使いやすいのだが，ファインダに親ディレクトリに移る機能がない，とか，キーボードでできることが限られているとか，アプリケーションの管理が面倒だ（バッチファイルやパスの概念がないから）という面もあった。これらはSystem 7.0で解決している。

特に，アプリケーションの管理はSX-WINDOWでもいえることだ。Macintoshでは，アイコンのダブルクリックで起動する。効率よく使うにはアプリケーションをひとところに集めておきたいが，そうすると，ひとつのフォルダにいろんなアプリケーションが混在することになって，非常にややこしい。かといって，アプリケーションを起動するたびにそのアプリケーションがあるフォルダのウィンドウを開くのは超面倒である。

System 7ではエイリアス機能を用意してスマートに解決した。アプリケーションだろうが，フォルダだろうがすべてのアイコンのエイリアスを作ることができるのだ。エイリアスは斜体で名前が表示される。エイリアスを使うと，自動的にその本体がアクセスされるため，アプリケーションのエイリアスをいくつも作っておき，データのあるフォルダに片っ端から突っ込んでおけば，いつでもそいつを起動できるようになるのだ。

System 7を使えない日本人の私は，しかたがないので，フリーウェアのAliasコマンドを使っている。On Cueという，メニューにプログラムを登録できるツールもある。

## QuickTime

Macintoshの未来を示唆する技術は2つある。ひとつはQuickTimeであり，もうひとつはPink OSである。さらにHyperCardを作ったビル・アトキンソンの会社，ジェネラルマジックが作っているP.I.C.マシンも忘れてはならない。

QuickTimeは，QuickDraw（クイックドロー）の動画版である。こいつが完成すれば，ワープロのドキュメントにアニメーションを付加することもできる。ワープロ文書を開くと，なぜか再生ボタンと停止ボタンのあるグラフィックがあって，そこで再生ボタンをクリックするとそのグラフィックが動きだすのだ。あらゆるアプリケーションで動画が再生できるようになる。

＊　＊　＊

Macintoshシリーズを眺めてみると，いままで多くのメーカーが作ってきたパソコンたちに比べてかなり無茶なことをやっている。キーボードの配列も時代によって変わっている。さらに，LCでは68020のデータバスを16ビットにして使っているし，Classic IIでは68030でそれをやっている。だから，Classic IIはClassicに比べてCPUは68000から68030へ，クロックは7.8から16MHzへとアップしているのに，実質的な処理速度は2～3倍にしかなっていない。普通，そんな設計はしない。しかも，IIsiではそれまでのIIシリーズすべてに積んでいたコプロをとってしまったので，一時期，動かないソフトが出てきてしまった。

Macintoshの上位互換性というのは，まったくもって完璧ではなく，Macintoshの作法に準じて作ったソフトだけが互換性を持つというものなのだ。しかし，ソフトハウスは新しいマシンが出て動かない部分が生じるとちゃんとそれに対応してきた。

Macintoshっていうのはそういう意味でも凄いマシンなのである。

**参考文献**

Macintosh，西林瑞夫著，共立出版

### Macintoshの常駐ソフト

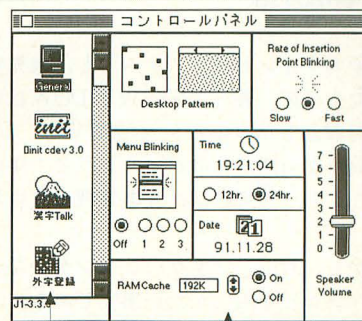
DOSやHuman68kでいう常駐ソフトやデバイスドライバは，Macintoshの場合，INIT/CDEVと呼ばれる。INIT（イニット）というのは立ち上げ時に自動的に読み込まれるソフト。CDEV（シーデブ）は，コントロールパネルデバイスの略で，立ち上げ時に読み込まれるだけでなく，コントロールパネルで扱えるものである。これらは常駐する。

DOSやHuman68kではこの手のソフトを常駐させるとき，CONFIG.SYSに記述するなり，AUTOEXEC.BATに記述するなり，手で起動するなりする必要があった。Macintoshはそうではない。システムフォルダというフォルダ（フォルダの名前はなんでもいい。要は，システムの入っている専用のディレクトリ）にINITやCDEVのファイルを放り込めば，次回から自動的にそれを読み込んで起動する。便利だけど，ときどきしか使わないINITはどうするか。シェアウェアや商品で，そういった問題を回避するものが作られている。そのCDEVを使うと，起動時に読み込むINITを選ぶことができるのだ（Macintoshにはシェアウェアやフリーウェアが欠かせない)。

INITやCDEVは非常によく使われる。私も，INITとCDEVを合わせて，10個以上常用している。アップル純正のシステムに添付されたものも加えれば，20個は下らない。すると，INIT同士のコンフリクトという問題が生じてくる。同じ資源を取り合ったり，IDがどこかで重なったりという現象だ。相性がよくない，というひと言で片づけることが多い（だって，追及するときりがないから)。それでも，みんな，多くのINITやCDEVを使う。

今回の図版でも，ファインダの図からいくつかのINIT/CDEVの存在を見てとれる（図1参照)。右上の時計はSuperClock!というCDEVだし，スクロールバーの矢印もScroll2というCDEVを使って変更してある。スクリーンセーバーもCDEVだし，リムーバブルハードディスクのローダーはINITだし，ウイルスチェック用のINITも使っているし，日本語FEPもCDEVだ。

さらに，日本語を表示するシステム自体がCDEVなのである。日本語版のMacintoshは，英語版のMacintoshに日本語を処理するCDEVを付加したものなのだ。

優秀なINITやCDEVはほとんどが欧米で作られたものであり，日本語のシステムと相性の悪いものもある。Macintoshのシステムはよく落ちる，といわれる。確かによく落ちるが，ユーザーが勝手にたくさん組み込んだINITやCDEVのせいであることも多い。システムに添付されたもの以外のINITやCDEV以外使わないことにすれば，けっこう防げるものだが（なかにはマルチファインダと相性の悪いアプリケーションというひどいものもあるが)，誰もそんなことはしない。いつ落ちるかわからなくても（再現性のない落ち方が多い)，快適な環境を選んでいるのだ。微妙なところである。Macintoshを常用し始めると，無意識のうちに“まめなセーブ”が身につくのは保証しよう。

NeXTの場合＝ユーザーインタフェイスを極める

# SX-WINDOWはNeXTの夢を見れるか？

Ushijima Takeo 牛島 健雄

Ver1.10発売以降，沈黙を続けるSX-WINDOWだが，多くのユーザーにとって関心の的であることは間違いない。NeXTのウィンドウシステムを参考に，SX-WINDOWの未来を考えてみる。

最近は夢を見なくなった。眠りが深いからだろうか，それとも睡眠時間が短いからだろうか。あるいは，単に忘れっぽくなったのだろうか。理由は何であれ，とにかく夢というものを感じられなくなって久しいのである。

いまの時代では，コンピュータの世界でも，夢のあるマシンが少なくなったといわれている。これは実感である。カタログを眺めていてもワクワクするものが感じられないのだ。

「なにかスゴイことができそうな予感」を感じさせるだけの雰囲気というか，それなりの意気込みが感じられないと思うのである。

それはそうと，我らのX68000が「夢を超えた」という看板を掲げて，意気揚々と出帆したのが4年ほど前。それから，いくつかの世代交代を重ねて，今は「父の電子手……いや，パソコンを超えろ」のXVIへと発展した。

いまや，キャッチフレーズに“夢”という文字は見当たらない。それでは，X68000に“夢”はなくなってしまったのか。

答えは否である。

X68000には“夢のウィンドウシステム”SX-WINDOWがあるじゃないか（なんて強引な展開なんだろうか）。

## 夢なかば

“夢のウィンドウシステム”と，大見栄きったものの，頼みのSX-WINDOWはいまだ発展途上なのである。本誌でVer.1.0の開発資料を公開したのが，ちょうど1年前ということを考えれば，まあ無理もないだろう。

しかし，対応アプリケーションが少ないのは，いかんともしがたい事実である。

こんなことをいっていると，なんとなく不安になってくるのだが，ゼロから出発したX68000の現在までの軌跡（奇跡かもしれない）を考えれば，これくらいはたいしたことはないということだ。

「少年よ大志を抱け」という偉い先生の言葉もあるくらいだから，現状を憂うより未来へと夢を膨らまそうではないか。

## NeXTなのである

さて，ここではSX-WINDOWの将来を語ろうというのだから，先達を決めなければ話は進まない。

「似てるんだからMacintoshを手本にするべきだろう」という人は甘い。Macは本来のマルチウィンドウではないし，夢のマシンとするには，いささか古いと思うのである。とはいっても，現状でMacから学ぶべきことは，ユーザーインタフェイスにおける面だけでも，かなりあることは周知の事実なのだが，あえて目をつぶることにする。

夢のマシンといえば，忘れてはいけないものがある。そう，NeXTである。

「NeXTは何故に夢のマシンであるか」という問いに解説を加えるのは，あまりに愚かな行為であろう。とりあえず，どこかで触ってみてほしい。実際に触っていただければ，自ずと答を見出すことができるだろう。

では，NeXTシステムの基礎を構成するソフトウェアについて，参考程度の説明をしよう。

## NeXT step

NeXTのアプリケーションは，NeXT-stepと呼ばれるマンマシンインタフェイスの上に構築される。

NeXTstepは，NeXTソフトウェアアーキテクチャのなかでもっとも重要なコンポーネントを統合して作られた作業環境で，ユーザーとアプリケーション開発者のどちらにおいても作業の土台となるものである。

図にNeXTstepを構成するソフトウェアを示した。

これらのなかで，InterfaceBuilderとNeXT WindowServerについて，その目的について簡単に説明しよう。

●InterfaceBuilder

アプリケーションのユーザーインタフェイスを視覚的に設計することと，プロジェクトごとに新たなプログラミング環境を作ることを主な目的とするソフトウェアである。実際には，アプリケーションのユーザーインタフェイスであるウィンドウを表示し，ボタンなどのインタフェイスオブジェクトを視覚的に操作できるようにするなどの機能がある。

また，インタフェイスオブジェクトの動作に関連を持たせ，新たなインタフェイスを作成することもできる。

●NeXT WindowServer

NeXT WindowServerは，画面に表示されるウィンドウを作成，操作する低レベルのバックグラウンドプロセスで，アプリケーションからの命令に従ってスクリーン上にイメージを描画することと，アプリケーションにユーザーイベントを送り返すことを目的とするソフトウェアである。イメージの描画にNeXT版Display Postscriptシステムを使っているのが特徴である。

＊　＊　＊

このほかのソフトウェアの役割や機能について詳しく述べていると誌面が埋まってしまうので，ここでは省略させていただく。関心のある方は，NeXTに関する資料をひもとくことをすすめる。ただし，紹介本の類は，さらに理解不能に陥る可能性があるので触らないほうが賢明だ。

## NeXTユーザーインタフェイス

NeXTの最大の特徴は，そのユーザーインタフェイスにあるといっても過言ではないはずだ。

NeXTユーザーインタフェイスは，初心者と経験者の両方のニーズを満たすものとして，簡潔性と効率性をともに達成させるために，次の4つの基本概念に基づいて設計されている。

1）インタフェイスが，すべてのアプリケーションにおいて一貫していること

2）ワークスペースと，そのウィンドウについては,ユーザーが自由に制御できること

3）インタフェイスが，ユーザーにとって自然な感じを与えるものであること

4）ユーザーとインタフェイスの対話の主な手段は，キーボードではなくマウスであること

これら，初心者に対して簡潔性を提供するためのインタフェイスが，そのまま経験者に対して効率性を提供するものとなっている。

アプリケーションは，ユーザーに対して，現実世界での操作に類似性を持ったインタフェイスを提供することで，あたかも日常の操作のひとつとして認識させることが必要なのである。

これは，コンピュータが独自の世界を構成するものではなく，われわれの世界を広げるためのオブジェクトとなることを示している。ここにおいて，NeXTは現存するコンピュータのなかで抜きん出ているのである。

これが，“夢のコンピュータ”と呼ばれる所以ではないかと思うのだ。

## どこまで近づけるか?

さて，NeXTの特徴がユーザーインタフェイスにあると述べたところで，われわれがNeXTから見習うべきことはなにか，ということについて考えてみたい。

**■ユーザーインタフェイスの充実**

ユーザーインタフェイスを向上させるためには，ボタンやメニューなどのインタフェイスアイテムを充実させることが必要不可欠である。現在のSX-WINDOWでは,お世辞にもインタフェイスアイテムが充実しているとはいえない。

たとえばメニューにおいて，階層化メニューをサポートするだけでも，操作系統の把握が容易になり,操作性は格段に向上する。

また，ウィンドウの形状についても，もっとシンプルなものを採用し，ミニチュアライズ機能を標準でサポートすれば，使い勝手はよくなるはずである。

アプリケーションの開発者が，ユーザーインタフェイスを構築するために頑張らなければならないのは，カッコよくないのである。

**■自然さの追求**

SX-WINDOWは，Humanを乗っ取るように起動されるためか，いつまでもCOMMANDシェルの環境を引きずっているようでよろしくない。

エディタひとつ起ち上げるにしても，いちいち起動ファイルを探してこなければ実行できないのでは，COMMANDシェルがウィンドウになっただけで，キーボードが使えない分だけ不便に感じられるのである。

そもそも，システムの機能が貧弱なのである。正確にはSXWIN.Xが，もっと高機能であることが必要である。

ファイルの検索をするために，わざわざ拙出のSXWHERE.Xを使わなければならないなどというのは，大きな間違いなのである。本来ならば，システムの機能として持っていなければならないのだ。

“文章を読みながら，傍らの辞書をひもとく”……これが，実現できてこそウィンドウシステムは「自然さを備えた」といえるのである。当然のことながら，NeXTでは実現されている。

SX-WINDOWシステムのなかでHumanをエミュレートするという形で，COMMANDシェルの呪縛を解き放ったところで，なんの問題もないと思うのは，あさはかな考えだろうか。

SX-WINDOWはCOMMANDシェルと地続きでないことをはっきりとさせないといけないのではないか。

**■表現力の拡大**

現在のところSX-WINDOWに，フォントを管理するためのマネージャはなく，グラフマンがすべて管理している。そのため，Macintoshなどに比べて文字の表現力がかなり弱い。

視覚的な自然さを求めるのであれば，まず第一に，文字について自然な表現ができなければならないのではないか。

そのためにも，NeXTに採用されているDisplay PostScriptを採用することを，個人的にすすめる。おりしも，いまはWYSIWYG大盛況の時代であるから，SX-WINDOWもこの波に乗らない手はないだろう。せっかくのグラフィック表現能力を無駄にするのは罰があたるぞ。

これだけの能力がありながら，DTPができないのは罪である。

**■センスを磨くべき**

NeXTに惹かれる原因には，外観からソフトウェアまで，“デザインのセンスがよい”ということがある。

とかく，日本人にないといわれるのが，この“センス”である。

このセンスについては，自分自身決してセンスがよいとは思っていないので，深く言及することはしないが……。

ウィンドウは視覚的な効果が顕著に表れるだけに，センスの悪さは，そのまま使いづらさに反映してくるので気をつけるべきだ。いや，SX-WINDOWのデザインのセンスが悪いとかいってるのではないよ。たぶん。

*　　*　　*

史上，もっともマニアック(?)なコンピュータであるNeXTと，発売当初はマニアックなコンピュータであったX68000，どちらもユーザーの思い入れは相当なものだろう。

それだけに，妥協は許されないのである。

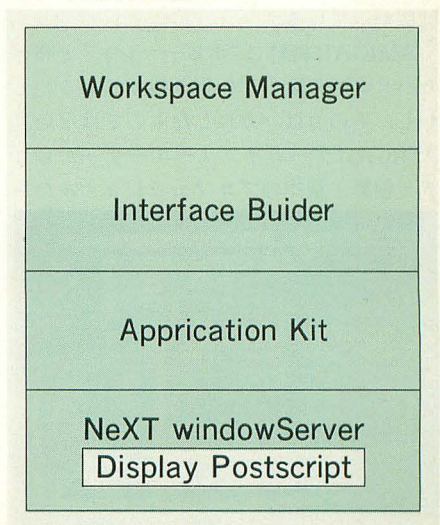
NeXTとX68000は，基本的なスペック，そしてコンセプトからして，まったく違う次元に存在しているのであるから，両者を比較してしまうのは無謀であるとは考えるが，ウィンドウ上の表現という点では，なんら問題がないはずだ。

SX-WINDOWは，いまだ成長段階にあるのだから，よいものはどんどん取り入れていくべきだろう。冒険ができるのはいましかないのである。失敗をおそれる必要はない。

MS-WINDOWSだってVer.3.0になって，なんとか使えるようになったし，OS/2だってVer.2.0になって使えるようになったみたいだ。

SX-WINDOWは，まだVer.1.1だ。長い目で，見つめていこうではないか。

**図1　NeXTstepの構造**

| Workspace Manager |
| --- |
| Interface Buider |
| Apprication Kit |
| NeXT windowServer<br>Display Postscript |

AMIGAの場合=破天荒なマシンには破天荒なウィンドウシステム

# 飛びそうに軽い,ワークベンチ

Akikawa Ryou 秋川　涼

いわゆる正統派のウィンドウとは別に，少し変わったウィンドウシステムもあります。AMIGAというマシンは本当のマルチタスクをやっていたり，ちょっと風変わりなパソコンです。ワークベンチはそういうマシンのウィンドウシステムなのです。

ウィンドウ環境といえば，MacintoshやWINDOWSの話が出てくるのが当たり前のようだ。しかし，ウィンドウ環境はなにもそこだけにしかないのではない。X68000にもビジュアルシェルとSX-WINDOWという2つのGUIがあるし，他機種でもウィンドウライクなシェルがある。

海外のマシンでも，MacintoshやIBM PCに限らず，ホビーマシンといわれるマシンにもウィンドウがバンドルされていたりする。ATARI STしかり，AMIGAしかりである。

まあ，これらのウィンドウは機能的にいえば，ウィンドウの王道とされている前述の2つのシステムにはひけをとるかもしれない。極端にいえば，ファイルセレクタともいえるものだから。

しかし，だからといって全然ダメというものではない。いろいろと欠点はあるものの，それなりに優れているところも多数見受けられるのだ。

ここではAMIGAのOSの話を例に取り，ウィンドウ環境のもうひとつの世界を覗いてみよう。決して洗練された世界ではないが，スリルと冒険心に満ちた世界がそこには広がっている。

AMIGAは純粋に“ホビーマシン”と呼べるパソコンの数少ない生き残りである。しかし，その力はバカにしたものではない。AMIGAはゲームオンリーユーザーからビデオ編集・制作のプロフェッショナルたち，強力なプログラマにまで使われている。富士山の裾野のように広いAMIGAの裾野。その広い裾野をカバーしているのがワークベンチ（WORKBENCH）といわれるAMIGAのGUIである。

## ワークベンチ=仕事場，細工台

ワークベンチを日本語にすると仕事場，細工台ということになる。実にぴったりフィットしたネーミングだと思う。ワークベンチでは，職人の仕事場のように荒っぽさがまかりとおっていて，きちきちに決まりごとが決められているわけでもない。ファイルに関する多大な管理もされていない。画面を見ていても殺風景で，かなづちの音や，棟梁の怒鳴り声がいまにも聞こえてきそうだ。

なにもかもが適当，といってしまえばそれまでだが，決して悪いことではないと思う。なんといっても，シンプル。使用メモリも少ない。電源を落とす前のウィンドウの状態なんて覚えてはいないし，そんなに凝った機能も特にないのだ。

つまり，ファイルを取り扱う最低限の機能だけが用意されていて，あとは自分でCLIなどを使って苦労するなり，ワークベンチを快適にするツール（パッケージで発売されているものや，シェアウェア，フリーウェアなどが多数ある）を組み込むなりして，“お前さん，適当にやっとくれ”という感じなのである。

ワークベンチの本体自体はメモリをほとんど消費しない。快適な環境を整備したいなら，いろいろツールを組み込んだりするだろうから，メモリはたくさん必要になってくるが，デフォルトの状態で使うなら，512Kバイトのマシンでもちゃんと動く。荒っぽいだけあって，飛んだりもするけど，Macintoshだって飛ぶもんね。

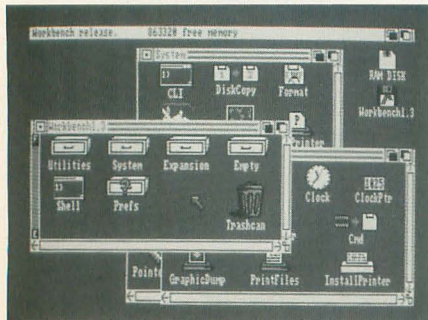
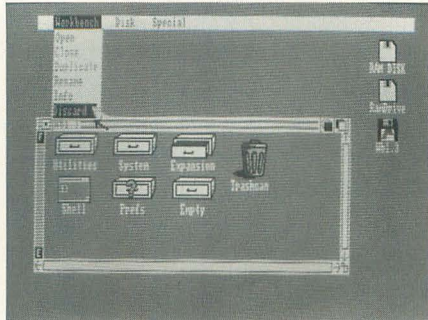
で，512Kバイトでも動き，本体はそれなりに小さいのだが，標準添付のシステムディスクはわりと容量いっぱいまで使われている。何種類かのフォント（ワークベンチのテキスト用ではなく，ワープロなどのアプリケーション用だが）や，時計，電卓などというお決まりのものはいいとして，ライブラリ，デバイスドライバ，プリファレンスなど，そして，なんとDOSコマンドまでディスク上なのは実に困ってしまう。

もちろん，実行ファイルのアイコンをクリックして作動させるくらいならディスクを読みにいくこともないが，ウィンドウ環境とはいえ，コマンドもよく使われるために，システムディスクへのアクセスは頻繁に行われることになる。

問題なのは，ディスクドライブ1台の場合。これは相当にまずいことになる。RAMがたくさんあってRAMディスクを確保できたり，ハードディスクを持っている場合なら，そこにシステムを転送してやればいいが，ないとするとディスク交換の嵐となってしまう。

これはもうウィンドウ環境がどうたらという騒ぎではなくなるのだ。ああ，悲しき1ドライブ。

## 立ち上がればもうウィンドウ

基本的に，ワークベンチはコマンドラインインタフェイス（AMIGAではCLIと呼ばれる）から呼び出されるかたちで立ち上がることになる。

シンプルなワークベンチの画面

プルダウンメニューもある

X68000でもHuman68kというものがとりあえず前提にあって，CONFIG.SYSに登録したり，AUTOEXEC.BATから呼び出すこととで，SX-WINDOWやVSが立ち上がる。WINDOWSもそうだ。

AMIGAでも立ち上げ一発目にはCLIが立ち上がり，startup-sequence(AUTOEXEC.BATと同じ働きをするテキストファイル）に書かれている諸々のセットアップを行ったあと，最後にLOADWBというコマンドでワークベンチが呼び出される。

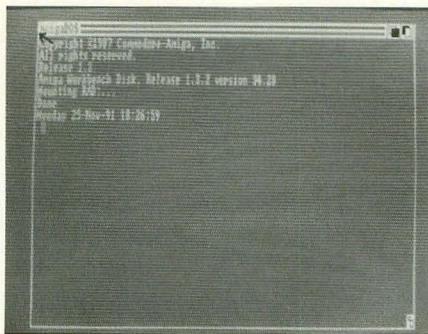
というわけなのだが，コマンドラインの手のひらの上にウィンドウが存在していて，指に小便をかける程度のことしかできない，という印象は受けない。なぜなら，コマンドラインとはいえ，それもすでにウィンドウの中で動いているからだ。わかりにくいかもしれないが，いかなる状態でもメニューバー（ワークベンチが立ち上がるまではメニューやドライブアイコンはなくて，ただのバーがあるだけだが）がついたフィールドは存在していて，そこに開かれたウィンドウの中でCLIが動いているということである。また，ワークベンチが立ち上がったあとで，CLIを呼び出した場合にもCLI用のウィンドウが開いて，その中で作業を行うことになる。

この差は大きい。もちろん，システムの組み方でSX-WINDOWなどやWINDOWSでも，最初からウィンドウ環境が立ち上がっているように見せかけることはできるが，コマンドシェルというものが別に存在するということを頭の中から消し去ることはできない（実に個人的な見解かもしれないが)。

AMIGAも最初はコマンドシェルしかなかったそうで，基本的には単なるDOSマシンなのだが，画面周りの配慮によって，純粋なウィンドウマシンでCLIというアプリケーション（シェル）も持っている（実際にはそうではないのだが）という印象を持てるのではないだろうか。

## startup-sequence

前述したように，ワークベンチでは電源を切る前の状況を覚えていたりはしない。だから，立ち上がったときにはウィンドウが何も開いていないということになる。

ここが最大の欠点だと思われる。最初から開いていてほしいウィンドウがある場合には，そういうツールを使うか，startup-sequenceに書き込まなければならない。startup-sequenceはSディレクトリ（各種スタートアップのセッティングなどが入っている）の中にある。

startup-sequenceもウィンドウ上で実行

CLIはいくつも開ける

メニューバーをつかんでスクリーンを下に

3つのスクリーンを同時に表示

が，そういうことができるのも，アプリケーションウィンドウに対してのみということになる。とあるドライブの，とあるディレクトリ(DRAWER)のウィンドウを開いておくということはできない。ウィンドウが立ち上がってから，ダブルクリックの連続使用で開いていくしかないのだ。

こうしたことや，環境の設定をstartup-sequenceに書き込まなければならないという作業は，面倒臭くて気分を損なわれるし，初心者にとってはわかりにくい操作ともなるだろう。このあたりには時代遅れな印象を拭い切れない。

## CLIもウィンドウの上

ワークベンチからCLIを走らせると専用のウィンドウが開く，ということは頭のほうで説明した。

まあ，普通のマシンならこの説明ですべて片付くんだけど，AMIGAの最大の特徴である，あるひとつの事柄が関係してきて，ただのコマンドラインのような働きではなくなってくる。

それはなにかというと，AMIGAはマルチタスクマシンということである。

ただのマルチウィンドウでもなく，イベントドリブン型の疑似マルチタスクでもなく，タイムスライスでタスクを切り替える，本当の意味でのマルチタスクである。

である，から，当然メモリがある限りCLIを何個でも開ける。何個でも開ける，ので，その上でプログラムを走らせても“はい，マルチタスク”，ということになる。

3つぐらいCLIを開いて，そこで同時に同一ドライブのディレクトリを取ったりすると，とても苦しそうだが，いちおう同時にやってくれる。

これは非常に便利である。便利なのだが，結構アブナイという気もする。

## 裏のスクリーン

というわけで，ワークベンチではCLIを含む，さまざまなアプリケーションが同時に動く。

でも，同じフィールド上で何個もウィンドウが開き，いっぱいアプリケーションが動いているという状況にはあまり，お目にかかれない。

ワークベンチのスクリーンは基本的には640×200ドットの大きさであるから，そんなにたくさんのウィンドウを同時に開けないことがひとつめの理由。そして，基本的には4色しか使えないので，ペイントツールなどをフィールド上のウィンドウの中で動かすことには無理がある，というのがふたつめの理由。おそらくこういった理由でフィールド上に開くアプリケーションが少ないのだと思う。

というわけで，ほとんどのアプリケーションはワークベンチのスクリーンの裏にも

Preferencesの設定画面

Infoメニューを選ぶ

う1枚スクリーンを開き，そこで走ることが多くなっている。

こういう場合，マルチタスクの意味がなくなってしまいそうな気がするが，そんなことは全然ない。

それぞれのスクリーン画面左上にデプス・アレンジメント・ガジェットがついているので，スクリーンどうしの裏表の入れ替えは簡単にできるからだ。

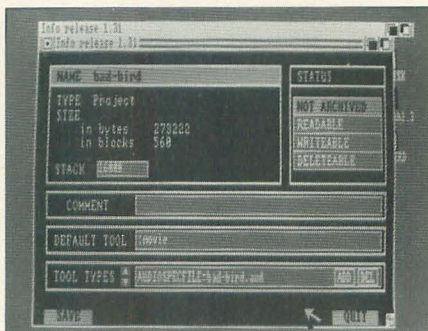
さらに，それぞれのスクリーンのメニューバー（見えなくなっているときもあるが，存在はしている）をマウスで下にドラッグしてもいい。そうするとどうなるのかというと，裏にスクリーンがある場合ならそのスクリーンが現れるのだ。

こういった機能のおかげで，別スクリーンを開くというよりは，画面いっぱいの大きさのウィンドウが何枚か開かれている，という感覚で各アプリケーション間を往来することができる。

また，ほかのマシンではウィンドウを何枚か開くと，前のウィンドウのパレットが変わってしまうということがあったりするが，スクリーンを何枚か開くという方法を取っているのでそんなことはない。スクリーンずらしでグラフィックモードの異なる何枚ものスクリーンを同時に表示させても見事に混在している。ノーインタレース，インタレースのスクリーンが混在することもある。その場合，ノーインタレースのスクリーンも走査線間を補間して，インタレースで表示しているようだ。

このメニューバーでスクリーンを下げて，下のスクリーンを見るという機能は，AMIGAの最大の特徴のうちのひとつだと思う。私などは人にAMIGAを見せるときには，いまだに必ずスクリーンを意味もなく上下運動させて，自己満足している。

## アイコン，ポインタを作る楽しみ

さて，ウィンドウといえば，やはりポインタやアイコンというものがつきものになる。ワークベンチでもメニューバー，ポインタ，アイコンの三種の神器を使う。

メニューバーには3つのプルダウンメニューが配置されている。

ひとつめはWorkbenchというメニューで，Open，Close，Duplicate，Rename，Info，Discardが用意されている。

2つめはDiskで，Empty Trash，Initialize。3つめはSpecialで，Clean Up，Last Error，Redraw，Snapshot，Versionがある。特に変わったコマンドはないと思うが，Infoは重要なのであとで触れる。

で，ポインタとアイコン。これは自分で自由にデザインできる。ポインタのカラーパレットはワークベンチで使っている4色とは別に，4色を設定するので表現力を制限されることはない。

アイコンのほうは当然ワークベンチで使っている4色を使うが，少し変わったことができるので，逆に表現力は高い。

つまり，アイコンの絵には2コマ使えるということである。ふだんの絵と，クリックしてアクティブにしたときの絵の2つ。

これをどういうふうに使うかというと，いちばん一般的なのがDRAWERのアイコンが引き出しになっていて，クリックすると引き出しが開いた絵になるというもの。

これはうまく使うと，元気なワークベンチを作ることができる。いろいろなアイコンをアクティブにするたびに，ワークベンチがダンスホールに様変わり。ちょっとクサい表現かな。

## Preferences，.infoで管理

ポインタを含むワークベンチの環境設定はPreferencesというプログラムでエディットする。そして，そのセッティングはDevsディレクトリ（デバイスドライバなどが入っている）にsystem-configurationというファイル名で格納されている。ワークベンチのパレット，解像度の設定，ポインタの設定，プリンタ，シリアルポートの設定などがPreferencesではエディットできる。

各アイコンは“.info”という拡張子のついたファイルで管理されている。つまり，“DPaint”というファイルは“DPaint.info”というファイルで管理されていて，アイコンの形状や，そのファイルの種別などが格納されている。

.infoファイルの内容は，対象アイコンをアクティブにしたあと，プルダウンでInfoというメニューを選べば，エディットすることができる。

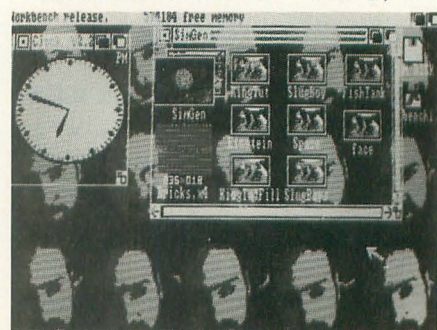
.infoにどんなパラメータがセットされているかは，写真を見ればおわかりいただけると思う。いろいろとあるが，いちばん重要なのは，“DEFAULT TOOL”というパラメータ。これはファイルのタイプがProject（なんらかのツールのデータファイル）のときにのみ存在するパラメータである。タイプがProjectであるアイコンがクリックされたときには，ここに書いてあるツールをオープンして，クリックされたファイルがそのツールのデータとして読み込まれる。

ファイルがドキュメントの類なら，テキストファイルの中身を表示するツール，アニメーションファイルであるのなら，アニメーションを表示するツールのパス名を書くことになる。

自分の.infoファイルがないファイルはワークベンチでは見えないことになる。これはなかなか不便である。コピーしたいときには，コマンドラインからやるか，ファイル管理のアプリケーションなどを使う羽目になる。

ちなみに，“picture.IFF”というような，拡張子つきのファイルの場合には，その.infoファイルは“picture.IFF.info”ということになる。結構変な感じ。

## 操作は全然統一されていない

AMIGAはマルチタスクマシンで，いく

背景に絵を置くSIMGEN

つかのプログラムが同時に走る。

もう何度もこのことは聞かされて，耳にタコができてしまった人もいるかもしれないが，マルチタスクなのは確かである。

しかし，同時にいくつかのプログラムが動くわりには，アプリケーションごとの操作法がMacintoshのように統一されてはいない。それどころか，まったくバラバラという感じである。

また，アプリケーション間のデータのやりとりをカット&ペーストで，ホイということもできない。

絵のデータなどをほかのアプリケーションへ持っていきたい場合には，RAMディスクに一度セーブするなりしなければならない。AREXXと呼ばれるREXX言語を使えば，こういうことも楽にできるのかもしれないが，普通は面倒臭いことをやるしかない。

まあ，幸い絵のデータのフォーマットなどはある程度共通化しているので，そこらへんは助かる。

## さまざまなハック

というわけで，不便なところと便利なところが混沌と同居しているわけなのであるが，不便なところというのは，ユーザー側ががんばることで徐々になくなっていくものと思う。もちろん，がんばるといっても，不便な操作に慣れてしまうのではなく，ユーザー側が便利なツールをどんどん発表していくということである。これはどんなマシンでもいえることだと思う。

メーカー側もワークベンチを細かくバージョンアップしているようなのだが，やはりユーザー側のほうが何も考えずにめちゃくちゃやっていて，面白いものを作っているようだ。

いちばん気に入っているのは，Arqというもの。最初に手に入れてファイル名を見たときには中身が全然想像できなかったので，たいして気にせず，しばらくほうっておいていた。しかし，組み込んでみるとなかなか面白い。

BASICで4つのウィンドウに出力

Arqとは“Animation Requester”の略で，「ディスクにライトプロテクトがかかっていて書き込めません」とかいったようなダイアログをメッセージだけで表示するのではなく，そのメッセージに合ったアニメーションを一緒に表示する，というものである。これは楽しい。わざとダイアログを出させたりしてしまう。

あとはアニメーションポインタとか，SIMGENとかいう，ワークベンチのバックグラウンドに背景の絵を合成するプログラムなどが面白い。

まあ，あんまりいろいろなプログラムを常駐させると，さすがにどこかでなにかがバッティングして暴走するので，使う側で気をつけなければならない。

実際に組み込んでいるのは，

**QMouse**　マウスにいろいろな機能を付加する

**CONMAN**　CLIにポップアップ，ヒストリー機能などを付加して，使いやすくする

**Open Look**　ガジェットの形を立体的なデザインにする

**Zoomwindows**　ウィンドウがMacintoshやVSのように，ズームアップされるような感じでシュパッと開くようになる

と，このぐらいだろうか。見た目のデザインが変化するものと，使い勝手がよくなるものの両方があるけれど，やはりどちらも必要だと思う。それはともに，ウィンドウを使う目的になっていることだからだ。

極端にいえば，かっこよく，見やすく，とか，操作をわかりやすく，単純に，ということが，ウィンドウという環境を作り上げる源になったのだろうから。

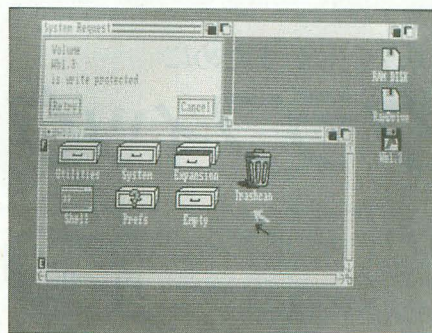
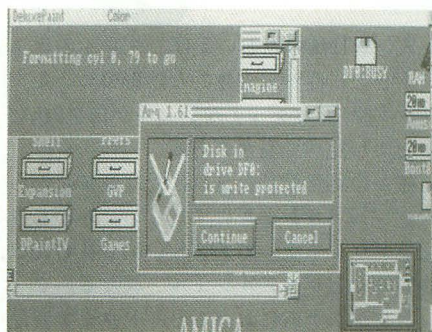
## BASICもあるよ

最後にBASICの話をしよう。ウィンドウ環境の話とはあまり関係ないかもしれないが，とりあえずご参考までに。

AMIGAには標準添付でBASICがついてくる。これもワークベンチの上で動くので，文字や図形などはウィンドウを開いてそこに出力することになる。リストもそうだ。

BASICから，

WINDOW 1,"Hitotume",(10,10)-(270,70), 15

というふうにすれば，window-IDが1で，Hitotumeという名前のウィンドウが，座標(10,10)に(260,60)の大きさで開く。15はウィンドウのタイプとなる。

これが通常のダイアログ

Arqを組み込むと，このとおり

もちろん，ウィンドウはいくつも開けるので，このあと，

WINDOW 2,"Hutatume",(310,10)-(580,70), 15

WINDOW 3,"Mittume",(10,95)-(270,170), 15

を実行すれば，全部でウィンドウが3つ開くことになる。

カレントウィンドウを選択するには，

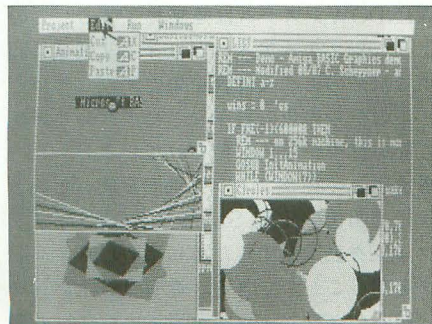
WINDOW OUTPUT 2

とすればいい。これ以降にLINE文でラインを書いたり，文字を出力した場合には，Hutatumeというウィンドウに出力される。

これなら，BASICでプログラムを書いてコンパイルできれば，簡単にウィンドウ上のアプリケーションも作れそうである。

＊　＊　＊

“便利なのだがアブナソウ”という表現を文中で使用したが決して悪い意味ではない。これはワークベンチ全体にわたって流れている空気であるから，あまり気にしないでおいたほうがいいのだ。ワークベンチを作ったメンバーも，使っているユーザーも，ワークベンチを便利にするツールを作るパワーユーザーも，たぶんそう考えているのだと思う。

安全で統一された，きれいなウィンドウ環境が本来あるべき姿なのだろうが，AMIGAはまったく正反対を行っているような気がする。そういうマシンなのだ。こういうのがひとつくらいあってもいいんじゃないだろうか。

GUIを生かすために

# グラフィック資源の管理と活用法

Tan Akihiko　丹　明彦

**GUIであるということ＝常にグラフィックを使用するウィンドウ環境でもX68000の強力なグラフィック機能がなくなってしまったわけではありません。ここではSX-XINDOWでのグラフィックの取り扱いと画面モードについて考えてみましょう。**

SX-WINDOWといえば，グレイのデスクトップに浮かぶ立体的な陰影のついたウィンドウだ。ビジュアルシェルが白黒2色だったのに比べて豪華である。

確かに外見だけを見ればただ豪華になったビジュアルシェルにすぎない。しかしウィンドウシステムであるSX-WINDOWは，本質的にビジュアルシェルと異なる。ウィンドウシステムにおけるハードウェア資源の管理には，通常のアプリケーションのそれより格段に高いものが要求されるのである。

SX-WINDOWは，ひとつのデスクトップの上にたくさんのディレクトリウィンドウやアプリケーションウィンドウを開くマルチウィンドウのシステムである。またSX-WINDOWは，デスクトップ上で同時に複数のアプリケーションプログラムを動作させることのできるイベント駆動型マルチタスクのシステムである。

そういう意味で，ディレクトリウィンドウだけを扱い，その上でアプリケーションを走行させるように作られていないビジュアルシェルとは本質的に異なる。

具体的には，それぞれのアプリケーションプログラムが占有するメモリの管理，それぞれのアプリケーションプログラムが操作するグラフィック資源の管理，そしてそれぞれのアプリケーションプログラムが動作するのに必要な実行時間の管理，などといったものをSX-WINDOWシステムが一元的に行っている。原則的にこれらハードウェア資源の管理の一切はSX-WINDOWシステムが行っており，アプリケーションプログラムがハードウェアを操作する際にはSX-WINDOWシステムを通さなくてはならない。ここが窮屈といえば窮屈であろう。

しかし逆に，ひとたびSX-WINDOWシステムにすべてを任せることに決めてしまえば，SX-WINDOWシステムの提供するあらゆるサービスを実に簡単に受けることができる。ウィンドウの生成や廃棄，移動やサイズ変更，ウィンドウ内部への描画，マルチタスク処理，コントロール（ボタンやスライドスイッチ）やポップアップメニューといったものの利用方法は公開されている。

そしてすべてのアプリケーションプログラムは一定の作法に従って制作されなくてはならない。作法というのは，先にいったようなSX-WINDOWシステムの利用方法は当然として，マウスボタンへの機能の割り当て方からウィンドウ上のプッシュボタンの配置の仕方にまで及ぶ。そうすることで初めて，アプリケーション操作性の統一という，ユーザー側からにとっての大きなメリットが生じるのである。

## テキスト画面

冒頭でも述べたとおり，SX-WINDOWは渋いグレイの色調（色相は赤系統，青系統などのように変更することもできる）で統一された立体感のあるウィンドウで構成された外観を持っている。NeXTと，ぱっと見にはよく似ている。

それを実現するのはビットマップテキスト画面を利用したグレイスケール4階調表示である。デスクトップ表示に調達されるのはテキスト画面。グラフィック画面ではない（ここでいうグラフィック画面とは，グラフィック表示のために設計された画面，テキスト画面とはテキスト表示のために設計された画面を指す。SX-WINDOWではどちらもいわゆるグラフィック表示のために使われている）。X68000のグラフィック周りの特徴のひとつとして，テキスト画面にドット単位で自由に操作することができるビットマップ方式を採用したことがあるのだが，そのメリットがここで生かされている。標準で768×512ドットと，比較的高解像でもある。ここにウィンドウをばしばし開く。

SX-WINDOWには，グレイ以外の色も使われている。

●赤青黄緑……比較的特殊な用途のために使われる色。たとえば画面右上にある，「X」と描いてあるアイコン。これはシステムアイコンといい，残りメモリ容量に関するステータスを表す。SX-WINDOWはマルチウィンドウ・マルチタスクのウィンドウシステムである。ウィンドウシステムはメモリを食う。ディレクトリウィンドウ，アプリケーションのウィンドウ，これらが一度にデスクトップ上に存在し，動作する。たくさんのディレクトリウィンドウを開き，たくさんのアプリケーションプログラムを走らせると，たとえ，その1つひとつの占有するメモリが少なくても，全体ではけっこうな量になり，限られたX68000のメモリ空間は残り少なくなってくる。

その具体的なメモリ残量（あと何Kバイト残っているか，といったこと）は「プロセス情報」ウィンドウで調べることができるが，メモリの残量はシステムアイコンにも反映する。それも「X」の字の色がメモリ残量に応じて変わる。メモリの残りが少なくなってきて，これ以上メモリを消費するとSX-WINDOWシステムの処理を無事に続けられなくなりそうだという事態に陥ると，「X」の字の色が通常の緑から黄へ，さらに赤へと変化する。ここまできてもなお不要なウィンドウを閉じないと，例のF1炎上のアニメーションとともにSX-WINDOWシステムは強制終了する。このようにシステムアイコンはメモリ残量に関する信号のような役目を果たす。

この信号の色の表示などに，グレイ系統でない鮮やかな色が使われている。

●透明……グラフィック画面を使いたいときに透明色を使う。SX-WINDOWは，グラフィック表示をするために透明なウィンドウを開くことができる。透明なウィンドウの後ろにグラフィックを表示すれば，そのグラフィックは透明なウィンドウを通して

画面に表示される。

透明は黒ではない。黒はそのさらに後ろにあるものを隠す。本当に真っ黒なウィンドウを開くことも可能である（あまりありがたみはなさそうだが)。このことでもわかるとおり，SX-WINDOWではテキスト画面の表示がグラフィック画面の表示に優先している。グラフィック画面を見るためにはテキスト画面を透明にしておく必要があるのである。

## グラフィック画面

そのグラフィック画面だが，通常の手段では見ることができなくなっている（透明なウィンドウを開かなくてはならない）ことからもわかるように，SX-WINDOWシステムの中ではそれほど重要な位置を占めてはいない。それでもグラフィック画面を利用する手段は正式にサポートされている。

標準の768×512ドットモードで同時表示できるグラフィックの色数は16色である。この16色は65536色の中から任意に選ぶ。当然テキスト画面を使ったデスクトップとは独立なので，好きな色を選ぶことができる。グレイスケールの落ち着いたウィンドウの中に鮮やかな色の絵が入っている姿は当初なかなかインパクトのあるものであった。

標準システムに入っていてグラフィックウィンドウを使うアプリケーションには，自転車に乗った暁子さんと犬がウィンドウの中を走る“暁子.X”と，PIX形式の画像を表示する“キャンバス.X”がある。とりわけ暁子さんは，SX-WINDOWのマルチタスク能力のデモンストレーションにもなっていたこともあり，SX-WINDOWのシンボル的存在であった。

ただ，表示する16色を65536色の中から選べるというのは，限られた色数を最大限に活用することができるという意味ではいいのだが，マルチウィンドウシステムとは多少相性が悪い。キャンバス.Xで同時に数枚の絵を表示して，そのそれぞれが異なる16色の組を選んでいた場合，アクティブウィンドウに表示された絵以外の絵の色はめちゃくちゃになってしまうのである。

## グラフマン

SX-WINDOWのグラフィック資源を管理するグラフマン(GraphMan)。グラフィックマネージャのこと。SX-WINDOWシステムを支える15のマネージャのひとつ。グラフマンは，アイコン，ウィンドウ，ダイアログ，メニュー，プッシュボタン，スライドスイッチといった高次のものから，文字，図形，画像といったプリミティブなものまで，描画にかかわるものすべてを一手に引き受ける。

グラフマンはさまざまなタイプのデータを扱う。当然，アイコンやウィンドウを描画するための道具はすべて備えている。直線，長方形，楕円，角の丸い長方形，文字列，ビットマップなど。

グラフマンは描画領域の管理にも優れている。SX-WINDOWはマルチウィンドウのシステムだから，グラフィックの描き換えが頻繁に発生する。ウィンドウの移動というちょっとしたことでも，デスクトップ(背景)やそのウィンドウと重なっているウィンドウなどに対して描き換えを行わなくてはならない。その描き換えに要する時間をできるだけ少なくするよう，アップデート領域を常に管理し（これはウィンドウマンの仕事)，そのアップデート領域の内部にしか描き換えを行わない（これはグラフマンの仕事)。

グラフマンは，ウィンドウ環境上で正しく描画している限りは，実によく面倒を見てくれる。アプリケーションプログラマとしてグラフマンと付き合うと，そのありがたみがよくわかる。いまいったアップデート領域の管理もそう。SX-WINDOWのほうで自動的にやってくれる。

アプリケーションがウィンドウを開く。その中に図形や画像や文字列などを描くとしよう（というよりほとんどすべてのアプリケーションウィンドウはそういうタイプのものである)。

このとき，アプリケーションプログラマは，アプリケーションウィンドウのほかのウィンドウと重なっていて見えない部分や，画面外にはみ出ていて見えない部分などが発生する可能性を考慮してプログラミングを行う必要はない。もちろん，それらは本来描画されるべきでない部分なのだから描画してもらっては困るのだが，それでもプログラマはわざわざプログラムしてやる必要はない。なぜならウィンドウの可視部分/不可視部分という情報はグラフマンが押さえていて（正確にはウィンドウマンが押さえていてグラフマンにその情報を渡す)，決してウィンドウの見えない部分，見えてはならない部分には描画させないからである。

したがってアプリケーションプログラマは，常にウィンドウの全体が見えているつもりでプログラムを制作すればよい。あとはウィンドウマンとグラフマンがよきにはからってくれる。これはアプリケーションプログラマにとって非常に楽である。ウィンドウの重なりを計算するというのは意外に面倒なものなのだ。

ウィンドウへの描画を通常のルートで行っている限り，グラフマンは期待どおりの動作をしてくれる。逆に，グラフマンを通さずに描画を行うことは禁止されている。

グラフマンの機能は豊富だが決して万能ではない。そこがウィンドウ環境に慣れていない人には多少窮屈かもしれない。面倒みのよくなった分，直接描画よりも速度が遅くなってもいる。それでもグラフィック資源の操作はグラフマンを介して行わなくてはならない。正しいウィンドウシステムとして，これは当然のことである。

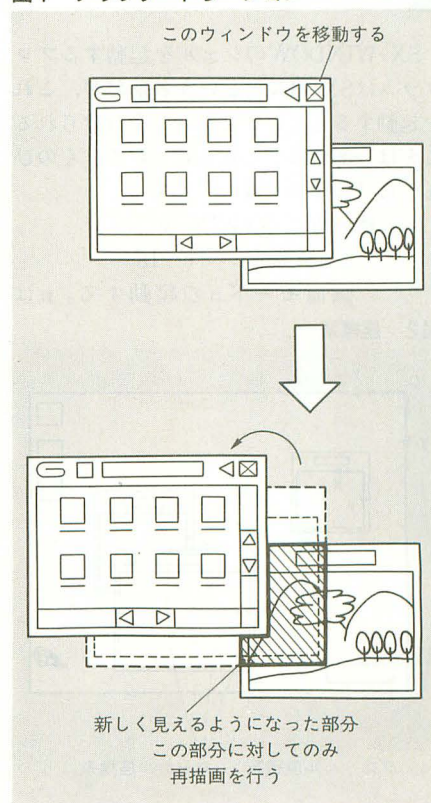
＊

グラフマンはX68000で使えるすべてのグラフィックモードを一応サポートしている。しかしSX-WINDOWで使われているのはそのうちのほんの一部である。また当面はその範囲に収めるべきだろう。

＊

グラフマン（とウィンドウマン）を特徴づけるキーワードのいくつかを挙げてみよう。

●リージョン……先ほどから数回出てきている「領域」という言葉に相当する概念。ウィンドウの見えている部分の形や，アッ

**図1 アップデートリージョン**

プデートの際に描き換える部分の形がこのリージョンに記憶されている。リージョンは複雑な形を扱える。

●ローカル座標とグローバル座標……ウィンドウはそれぞれローカル座標系という固有の座標系を持っている。ローカル座標はウィンドウの左上を原点(0,0)とする座標系で，そのウィンドウがデスクトップのどこに置かれているかということにはまったく関係ない。アプリケーションプログラマはこのローカル座標を基準に描画を行うだけでいい。

実際にグローバル座標を計算して画面に表示するのはグラフマンのほうでやってくれる。グローバル座標系はごくふつうの座標系で，画面左上を原点とする。

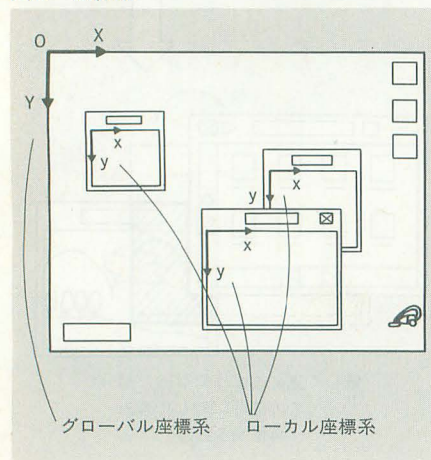
●スクリプト……描画の手順を記録するもの。ウィンドウへの描画はパラメータを設定して描画関数を呼び出すということを次々に繰り返すのがふつうだが，その手順を一定のフォーマットでスクリプトに記述することもできる。たとえば背景の模様を描画する場合などのように比較的パターン化された描画にはスクリプトが使われる。

スクリプトを用意しておけば，グラフマンのほうでそれを解釈して描画する。スクリプトを上手に使えば，プログラミングの手間を省くことができるだろう。

## 使いこなす

SX-WINDOWのシェルを起動するプログラムはSXWIN.Xという名前だが，これを起動するときにオプションを付けられることは一部ではよく知られている。そのひとつに「-gオプション」がある。

使い方： SXWIN -g [n]

n = 0~18

画面モードnで起動する。nはIOCSコールCRTMODに与える引数に対応している。

(例) SXWIN -g16 …… 768×512ドットモードで起動する（デフォルト）

SXWIN -g18 ……1024×848ドットモードで起動する（ただし24kHz表示のできるディスプレイが必要）

768×512ドットモードが比較的高解像とはいっても，やはり何枚もウィンドウを開くには狭いので，－g18オプションはありがたい。ただし24kHzのディスプレイが必要なことと，ちらつくことが欠点ではある。

さらに「実画面モード」で起動するオプションも存在する。実画面はあまり意識されていないが，X68000は1024×1024ドット分のテキスト画面を持っており，ディスプレイにはそのうち一部だけを表示する。通常の768×512ドットモードでは，X68000のグラフィック資源は半分も利用されていないことになる。これをSX-WINDOWで目一杯利用できるようにするのが実画面モード。実画面モードで起動すると，ウィンドウなどの描画を1024×1024ドットの実画面いっぱいに行うようになる。

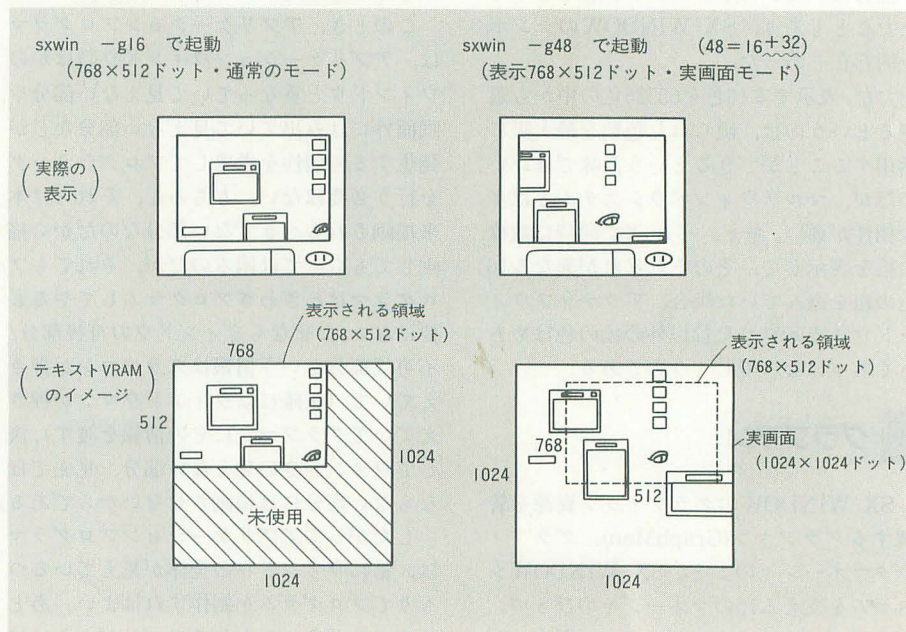
使い方： SXWIN -g [n]

n = 32~50

画面モード(n-32)で起動する。

(例) SXWIN -g48 …… 表示768×512ドットの実画面モードで起動する

SXWIN -g50 …… 表示1024×848ドットの実画面モードで起動する（やはり24kHz表示のできるディスプレイが必要）

同じ-gオプションなのだが，通常のモードを指定するのに用いるスクリーンモードの番号（-gの後ろで指定する数字）に32を足したものを用いるだけでよい。

ただし，実画面モードにしただけでは実画面全体を有効利用できない。マウスカーソルは実画面全体を動けはするが，ディスプレイに表示されている範囲から外に出ると見えなくなってしまう。したがって，依然としてディスプレイに表示されている範囲でしか作業できない。

ここは公開ソフトに頼らざるをえないのが現状だ。その公開ソフトにはいくつかあるが，いずれにしてもこれを組み込んでおけば実画面全体で作業ができる。マウスカーソルを画面の端に移動すれば画面がスクロールするので，実画面のすべての範囲で作業ができるようになる。ディスプレイに一度に表示されるのは768×512ドットでも，論理的レベルではメガピクセル（1024×1024=1M）の世界。

いずれにしても，デスクトップの一部しか見えていないという意味で，実画面モードはSX-WINDOWをある程度理解したユーザーが納得づくで使うものであろう。

## 私事ですがSXIMAGEバージョン2.0の予告

SXIMAGE.Xは，1991年5月号の付録ディスクに載せたSX-WINDOW上画像ファイル表示アプリケーションである。X68000の世界でポピュラーな画像形式であるPIC形式とCUT形式をサポートし，その点でだけキャンバス.Xより利用価値があった。

PICファイル表示はカラーなのだが，グラフィック画面のパレットに標準パレット

図2 座標系

図3 実画面モード

を導入したので，複数の画像を表示しても色に矛盾が生じないという特徴がある。ただPICファイルの処理が非常に遅いというほか，数々の欠点を抱えていたので，デバッグも含めたバージョンアップ作業を現在行っている。

現時点までに実現したものは次のとおり。

●PICファイルの処理がSXIMAGEバージョン1.0よりずっと速い。2倍は確実に出ているだろう。

●PICファイルのローディングをマルチタスクで行うようにした。これにより，ユーザーが待たされる時間はSXIMAGEバージョン1.0に比べ大幅に減った。処理そのものを高速化したということと合わせて，かなり快適になっている。

●画像をデスクトップ(背景)に敷くことができるようになった。背景描画は前述のとおりスクリプトで行っているのだが，それをコントロールする方法を最近覚えた。

より多くの画像フォーマットに対応することも検討している。というよりも，新たな画像フォーマットに対応する作業を行いやすいようにソースプログラムを書き，拡張方法とそれに関するガイドラインを示そうと考えている。

## 最後に

X68000のグラフィック環境は豊かである。その上に構築したSX-WINDOWのグラフィック環境には将来性がある。ただ早めに解決しておくべき問題がある。それは，

●アプリケーション作成のガイドライン
●使い回しのきくデータフォーマット

を確立することである。

SX-WINDOWアプリケーションの開発環境も徐々に整い，今後ユーザーからいろいろと便利なアプリケーションや楽しいアプリケーションが出てくることであろう。ウィンドウ環境の価値を決めるのは，こうしたアクセサリの充実度であると思う。

そのとき，操作法がまちまちだったり，扱うデータフォーマットに全然互換性がなかったりすれば，一度に複数のアプリケーションウィンドウを開いて作業を行うことのできるウィンドウシステムの価値が半減してしまう。

Macintoshはこのあたりに関して実に厳しく，Macintoshが成功したのはそのためではないかとも思う。先日Macintoshを使っていて，試しにMacDrawからカットしてきた図形のデータをワープロの上でペーストしてみたらあっさりと張り込めてしまった。そして僕は，不覚にも「こやつできるな」などと感心してしまった。本当は感心するほうがおかしいのである。

たとえばワープロには，中途半端な図形編集モードを持たせるよりも，優秀な図形編集ソフトのデータをそのまま取り込める機能を持たせたほうがDTPツールとして賢いし，使いでがあるし，プログラムサイズも小さくなるし，ユーザーが余計な図形編集モードの操作方法を覚えなくてもいいし，と，いいことずくめである。ただしそれを実現する環境には，次の条件が必要だ。

1) マルチウィンドウ/マルチタスク(疑似マルチタスクでもいい)であること

2) アプリケーション間のデータ互換が完全に取れていること

3) アプリケーションの操作スタイルができるだけ共通のものであること

SX-WINDOWを見ると，1)はすでに満たし，3)はまだまだアプリケーションが少ないからこれからのこと，そして肝心の2)については先行き不安である。まさにアプリケーションが少ない今が最後のチャンスかもしれないのである。

幸いグラフマンにはひと通りのグラフィックプリミティブが用意され，それを描画するための手続きもスクリプトの形で提供されている。素晴らしいSX-WINDOW環境の明日は，これを尊重することから始まるのかもしれない。ここをやりそこなうと，Macより出てMacに及ばずという情けないことになってしまう。

Maciontoshの世界ではこれまでの静止画像に加えて，新たに動画像も扱えるフォーマットを構築していると聞く。そのうちイメージワープロで文章の中に張り込まれた図版がアニメーションしているという状況が出現するのだろうか。

### /G18を使う

さて，X68000で1024×848ドットの高解像度モードを表示するためには24kHzのインタレースモードを使用しなければならない。一部の標準ディスプレイはこれに対応していないし，対応していてもちらつきが激しくて長時間の使用には耐えない（ディスプレイによってかなりの個体差があるようだ）。

そこで超残光ディスプレイを使用することになる。今回はエプソンのCR-5500で実験してみた。これは長残光の蛍光体を使用している。すでにお気づきのように，今回の画面写真のほとんどはこのディスプレイと高解像度モードで撮影されている。

元々エプソンの高解像度国民機用のディスプレイで，マルチスキャンでX68000の使用する15k〜31kHzをもカバーしているものだ。大きさは15インチ。17インチのCR-7000も発売されている。コネクタはJISの15ピンアナログRGBだから接続に際しても特に注意することはなにもない。

長残光というのは，カーソルを動かすとすーっと影が尾を引くタイプのディスプレイだ。アクションゲームはしないほうがいいかもしれない。

SX-WINDOWを立ち上げてみる。

インタレース走査のため画面は少し暗く，カラーの色調は悪くなる。しかし，画面を少し暗めにしておけば，ちらつきの大部分は除去される。実用になるレベルである。

X68000の標準ディスプレイと同じ15インチ管を使っているにもかかわらず，PC-286用に調整されているためか垂直振幅が足りない。後ろのつまみで縦幅，横幅，表示位置の調整が可能だが31kHz時と24kHz時で再調整が必要だ。

ほかにもソニー，三菱などがマルチスキャンタイプの中・長残光ディスプレイを発売している。また，理論上，最近は非常に安く入手できる場合もあるVGAモニタ（中残光）も使用できるかもしれない。

*　　*　　*

AMIGAではインタレースモードを多用する場合がある。ちらつきをなくすために「フリッカーフィクサー」と呼ばれる周辺機器が市販されている。（同名のアクリル板もあるが……）ケーブルの途中につないでちらつきを防止するハードウェアだ（高解像度モニタが必要）。

驚いたことに，同様なことを行うソフトウェアもあるらしい。具体的になにをやっているかは不明だが，多分ちらつきの出にくい色づかいにするものだろう。また，理屈では画面を構成するドットが縦2ドット組で使用されていればちらつきは発生しない。画面のほとんどをそのような方法で描画することでちらつきを抑えるものもあるようだ。

ウィンドウでのプログラム
# これからのプログラミング環境

SX-WINDOW

Nakamori Akira 中森 章

マルチウィンドウであること，マルチタスクであることはプログラムを作る環境としても優れています。エディットからデバッブまで含め，SX-WINDOXでのプログラミング環境について考えてみましょう。

## SX-WINDOWの現状

X68000で利用できる本格的なウィンドウシステムとしてSX-WINDOWは登場しました。複数のウィンドウを（疑似）マルチタスクで並行動作させるこのウィンドウシステムは，アイコンによるコマンド起動システムとして従来から知られていたビジュアルシェルとは一線を画すものとして私たちに受け入れられたのです。

ビジュアルシェルにしろSX-WINDOWにしろ，ウィンドウ内にあるアイコンをマウスでクリックすると，それに対応するプログラムが起動されて実行されます。このとき，ビジュアルシェルではいったんHuman68kに戻ってから，そこでプログラムが実行されますが，SX-WINDOWでは新しいウィンドウが開いて，その中でプログラムが実行されるというのが大きな違いだったのです。

ただし，これはSX-WINDOW用にあらかじめアプリケーションプログラムを作ってある場合に限ります。昔ながらのプログラムを実行させると，SX-WINDOW上でもHuman68kに戻ってしまうことはビジュアルシェルと変わりありません。つまり，なにか実用的なことを行わせる場合，ビジュアルシェルもSX-WINDOWも大差はありませんでした。

結局，SX-WINDOWは「目玉」や「暁子さん」や「時計」などをマルチタスクで動かして眺めるだけの環境ソフトにしかすぎなかったのです。SX-WINDOWが登場してからも，私たちは依然としてHuman68kの環境でプログラムのソースコードを書き，コンパイル（アセンブル）し，アプリケーションを実行していたのです。

しかし，ビジュアルシェルとSX-WINDOWではその存在意義が異なっています。ビジュアルシェルは，単にマウスでアイコンをクリックしてコマンドを実行したり，ファイルをコピーするだけのものでしかありませんでした。一方，SX-WINDOWはウィンドウ上でプログラムを実行するための手段，つまり，システムから抜け出さず（Human68kに戻らず）にコマンドを実行するための手段が提供されていたのです。ウィンドウやメニューなどの便利な機能をアプリケーションで利用できるという可能性だけはあったのです。

とはいえ，シャープからの技術資料の公開が遅れ，そのドキュメントも難解を極めていたところから，SX-WINDOW用のアプリケーションを作る人も少なく，できあいのアプリケーションを眺めるだけのウィンドウシステムという状況がしばらく続いたのでした。その間にSX-WINDOW用のアプリケーションもいくつか作られるようになりましたが，それらのほとんど（すべて？）はHuman68k上でソースコードが書かれ，コンパイル（アセンブル）され，リンクされたものだったのです。

SX-WINDOWの転機はSX-WINDOW上で動作する「エディタ」の登場です。これにより，プログラムを作る場合，少なくともソースコードを書く作業だけはSX-WINDOW上で行えるようになったのです。ほかのプログラムのソースコードや各種のドキュメントを参照しながらソースコードを書く必要があるとき，それらのファイルを個々のウィンドウとしてオープンして自由に参照することができます。邪魔になったらマウスで簡単に別の場所に移動してやればよいのです。

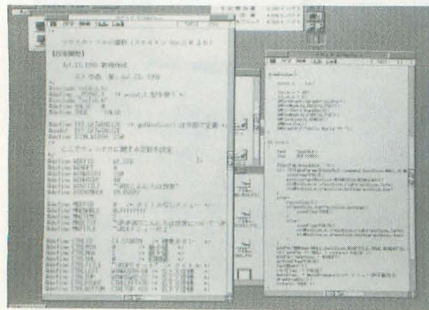
エディタ.Xを使う

また，カット・アンド・ペーストの機能を使用すれば，ほかのソースコードの必要な部分を自在に取り込むことができます。また，複数のドキュメントの切り張りが自由に行えるということは，プログラムの作成だけでなくなにか文章を書くときにも有用でしょう。

確かに，「エディタ」以前にもSX-WINDOWには文章を編集するための「ノート」というアプリケーションはありました。複数のウィンドウを用いたファイルの同時参照，カット・アンド・ペーストはそのときでも可能でしたが，それは，本来の編集機能が充実していることが前提です。ほんのメモ代わりに使うのならともかく，「ノート」を使ってそれでプログラムを書くという気にはなりませんでした。

とにかく，「エディタ」の登場によって，SX-WINDOWが開発環境としての第一歩を踏み出しました。いまはソースコードを書くためのエディタしか存在しませんが，アプリケーション作成の大部分の時間はソースコードの作成です。この時間をSX-WINDOW上の「エディタ」で費やすことで，私たちのプログラミングの基盤も徐々にSX-WINDOWへと移行していくのではないでしょうか。その場合，SX-WINDOW上にどのようなソフトウェアツールが存在することが必要なのでしょう。このことについて，以下で考えてみましょう。

## Macintoshのプログラミング環境

SX-WINDOWは外見はNeXTを，内部はMacintoshを真似たウィンドウシステムです（どちらもスティーブ・ジョブズが創ったものですね）。Macintoshそのものとまではいかないまでも，Macintoshのシステムに多大な影響を受けていることは，SX-WINDOWの技術資料を読めば一目瞭然で

す。SX-WINDOWでのプログラミング環境を考える場合，まずMacintoshのプログラミング環境がどうなっているのかを知っておく必要があります。

いまさら，いうまでもないことですが，Macintoshは，オーバーラップする伸縮自在な複数のウィンドウ，手軽なプルダウンメニュー，マウスとアイコンを主体とする操作といったユーザーの視覚に直接訴えるインタフェイスを特徴としているパソコンです。その上では，開発，実務，アート，ホビーなど，ありとあらゆる分野に対応した高機能で使いやすい多くのアプリケーションを利用することができます。

手軽に使えるアプリケーションがあまりにも多いので，Macintoshでプログラミングをしてやろうと思う人はまずいません。というか，Macintoshは，パソコンというよりも，いろいろな作業を可能にする「道具」という認識が強いのではないでしょうか。プログラムを書くということは，パソコンを自由に操ってやろうという意志の表れです。すべてが至れり尽くせりのMacintoshに対してそんな感情を抱く人がいるとはあまり思えません。

とはいえ，Macintoshのアプリケーションを作ってユーザーに提供する側は確かに存在します。このような人たちはどうやってアプリケーションを開発しているのでしょう。Macintoshのシステムには，わがままできまぐれな人間に柔軟に対応できる高度なインタフェイスを持ったシステムであることが要求されます（そうでなければ，誰もそれを使ってくれない）。

これは，プログラマにとっては大変な負担になります。Macintoshでは統一的で優れたユーザーインタフェイスを実現するためにツールボックスと呼ばれるサブルーチン群をあらかじめ用意してあります。これは，具体的には，高速のグラフィック描画，文字フォント，イベント，ウィンドウ，メニュー，テキスト編集，対話ボックスなどのインタフェイスを実現するのに必要な機能を持ったサブルーチン群で，Macintosh内部のROMに格納されています。

このツールボックスを利用することでプログラマの負担を少しでも減らそうとしているのです。実際のところ，このツールボックスを利用しなければMacintoshでのプログラミングは不可能といってよいでしょう。ところが，このツールボックスを利用したとしても，Macintoshでのプログラミングは非常に難しいものとされているのです。

これは，イベントドリブン，ノンプリエンプティブなマルチタスクという従来の（ウィンドウシステム以外の）プログラミングとは180度転換した考え方を要求されるためですが，これについてはもう少しあとでお話ししましょう。

さて，Macintoshでのプログラミングは，従来の方法とはまったく別の考え方をしなければなりません。しかし，実際の（実行形式の）プログラムの作り方は従来と同じです。すなわち，エディタでソースコードを書き，それをコンパイル（あるいはアセンブル）し，リンカでツールボックスのライブラリを結合して，実行形式のプログラムができあがるわけです。

しかし，プログラムを作成する環境においてまでも「すぐれたユーザーインタフェイス」というMacintoshの特徴はついてまわっているようです。

Macintoshの開発元のアップル社は，Macintoshシリーズの公式な開発環境としてMPW（Macintosh Progurammer's Workshop）なるものを供給しているそうです。その実体はコマンドインタプリタ機能つきのエディタを中心とするコンパイラなどのソフトウェア開発用ツール群ということですが，私は実物を見たことがないので詳細を説明することはできません。MPWを使えばどんなこともできる半面，操作が難しく，しかも高価で初心者向けではないという評判です。

一方，最近本屋さんの棚を賑わせているのがTHINK C，THINK PASCALに関する書籍類です。THINK CやTHINK PASCALはシマンテック（Symantec）社が開発したMacintoshのもうひとつの統合開発環境です。この環境は安価（MPWの1/4程度）であるにもかかわらず，ひととおりのツール類が揃っているということで注目を浴びているようです。

THINK C（あるいはPASCAL）の特徴をひと言でいうと，UNIXのmake（XCのver.2.0にも付属していますね）の機能を備えたエディタを中心とする，リソースエディタ，コンパイラ，ソースコードデバッガなどの開発ツール群ということができそうです。こちらも環境自身はMPWと大差ないもののように思えます。

なお，THINK CにはANSIライブラリという，標準入出力をウィンドウでサポートするためのライブラリが付属していることが特徴です。これはK&Rの第2版で紹介されている（ANSI C準拠の）プログラムをそのまま動作させることを目的としたライブラリであると考えられます。

Macintoshにはこのほかにも多くの開発環境があるようです。しかし，MPWやTHINK Cの環境から推測すると，結局のところ，Macintoshの開発環境は，通常のアプリケーションと同じ操作性を持ったウィンドウ上で動作する，賢いエディタ，コンパイラ（アセンブラ），デバッガとツールボックス以外のライブラリの充実というところに行き着くと思われます。

## SX-WINDOWに望む開発環境

前の節で触れたMacintoshでのプログラミングについての説明はそのままSX-WINDOWにも当てはまります。ツールボックスに相当するサブルーチン群はFSX.Xというデバイスドライバに格納されています。私たちはFSX.Xのルーチンを，これまたMacintoshと同じCPU（68000）のA系列の未実装命令を利用して，呼び出すようになっています。

イベントドリブンなマルチタスクを意識したプログラミングを強いられるということも同様です（ということは，プログラミングが難しいということ）。

### 楽勝プログラミングのタネ…TPWの場合…

最近，MS-WINIOWS3.0上では手軽にプログラムを組むためにさまざまな言語が登場しています。なかでもボーランド社から発売になったTurboPascal for Windows（以下TPW）の機能には目をみはるものがあります。TPWでは非常に簡潔にプログラムを書くことができ，しかもTPWからはWindwsの持つサービスをほどんどすべて使うことができるのでしょう？　それはTPWに付属されるobjectWindows（OWL）というクラスライブラリに秘密があります。

このライブラリには簡単にいうならば，たとえばウィンドウとは窓が開くものである，中は真っ白である……「ウィンドウかくあるべし」という一般的な動作があらかじめ定義されているのです。そこでユーザーはそれを“継承”して，OWLに書かれた動作とは違った動きをしてほしい部分，たとえばeyesのようなアプリであればウィンドウ内に目玉を描いてマウスにあわせて動かせといった，いわばプログラムの本質の部分だけ新たに定義してやれば，それ以外の部分はなにもしなくてよくなっているのです。

これを実現している“継承”は，TPWに限らずC＋＋などの言語に特徴的な機能でもあります。X68000にもg＋＋などがあることですし，こんな環境がSXにも登場するといいですのね。

（で）

さて，MacintoshとSX-WINDOWの違いはなんでしょう。考えるまでもありません。Macintosh用には使い勝手のよくて高機能なアプリケーションが山ほどあるのに，SX-WINDOW用にはほとんどないということです。結局，「自分でプログラムを作らねば」という気分にさせてくれるのがSX-WINDOWなのです。

よくも悪くもSX-WINDOW（が動作しているX68000）は「道具」ではなく，「パソコン」ということなのです。このため，Macintoshでは一部のアプリケーション作成者のための秘密の開発環境でしかなかったMPWやTHINK Cが，SX-WINDOWでは当たり前の開発環境として必要になってきます。

現時点でSX-WINDOWにはどのような開発環境があるのでしょうか。エディタはあります。しかし，それはシェルの機能を持ったものでもmakeの機能を持ったものでもありません。ごく普通のエディタです。しかし，それでもないよりはマシというものです。

まずは，コンパイラ（アセンブラ）やリンカなどシステムから抜け出すことなく使えるひととおりの開発ツールが必要でしょう。それがないことには話が始まりません。ウィンドウ上で動作するこれらのツール類はどのような形態になるのでしょう。

ウィンドウシステムで動作するコンパイラやリンカとしては，ファイルのアイコンをマウスでつまんでウィンドウに放り込むだけで，指定したファイルのコンパイルなりリンクが行われるのは当然です。

また，マルチタスクというからには，コンパイルやリンク中にほかのアプリケーションの動きが停止するなんてもってのほかです。複数のファイルを同時にコンパイルすることができてもよいでしょう。私自身がいまほしいと思っているSX-WINDOW上のコンパイラ，アセンブラ，リンカは次のようなものです。

・ウィンドウの外観は図1のようになっています。もっとかっこいいものになればそれにこしたことはありませんが。

・一括してコンパイルしたいファイルのアイコンをまとめて選択してウィンドウに放り込むと，最初のファイル名がウィンドウに表示されます。このとき，コンパイルオプションは規定値が設定されます。

・ファイル名，コンパイルオプションはキーボード入力で変更することができます。もちろん，ファイルをアイコンで指定せずに，ファイル名やコンパイルオプションをいきなりキーボードから打ち込んで指定することも可能です。

・取り込んだファイルの一覧はマウスのなんらかの操作（たとえばポップアップメニューで選択することによって図2のようなウィンドウが開く）で参照・変更ができるようになっています。また，その中で実際にコンパイルするファイルを個別に指定することも可能です。

・コンパイルを始めると個々のファイルのコンパイルの進行状況をピークメーターなどで知ることができるようになっています（これでコンパイラが生きていることがわかる）。

・もし，コンパイル中にエラーが発生すると「エディタ.X」が呼び出されタグジャンプできる形式のエラーメッセージが表示されます。エディットを終了するとコンパイルが再開されます。この機能は真里子版GCCに備わっているのと同様の機能です。

・簡単なmake機能が付いています。すなわち，前回のコンパイル時より変更のあったファイルだけがコンパイルされてリンクされます。

図1

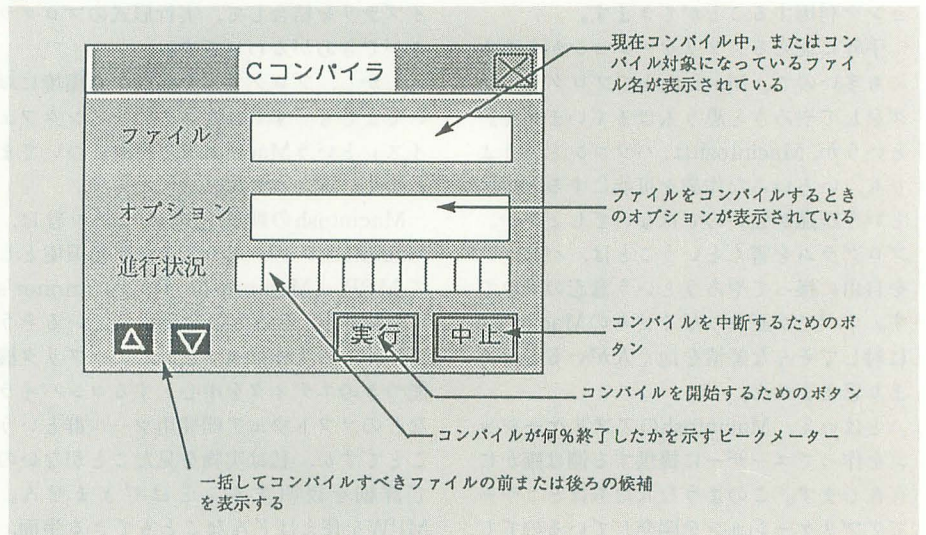
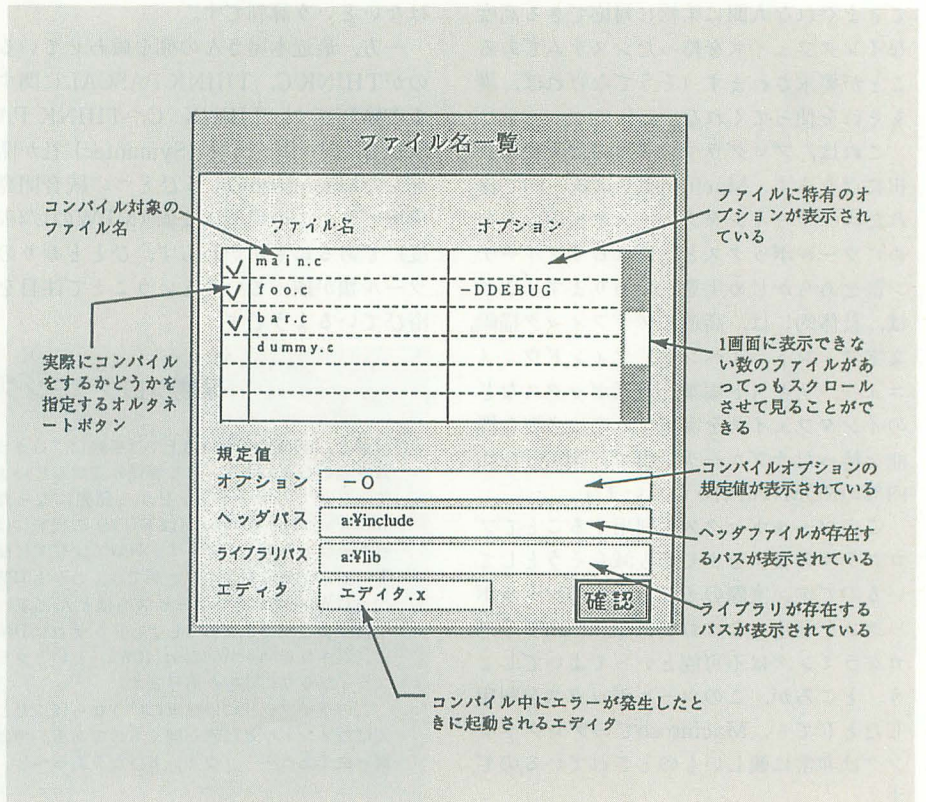



図2

・このウィンドウはマルチタスクで動作することが可能になっています。

＊　＊　＊

ざっと，以上のようなイメージですが，いかがでしょう。まあ，コンパイラ，アセンブラ，リンカ単体としてはこんなものでいいのではないかと思っています。

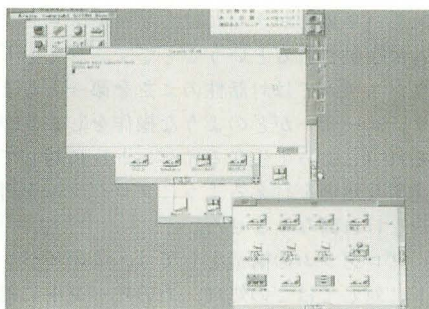
しかし，重要なのはコンパイラなどを起動する環境のほうでしょう。コンパイラなどとは別に本格的なmake機能を実現するようなウィンドウがぜひともほしいところです。アプリケーションの作成はコンパイラだけで行うものではありません。ソースファイルやデータファイルを，なんらかのツールで変換して新たなソースファイルを作成してから，それをコンパイルするという場合もありえます。完成したファイルをインストールするために適当なディレクトリにコピーする作業が必要な場合もあります。このような，ファイルをコンパイルするという以外の作業を，あらかじめ指定されたファイルの依存関係を調べながら，自動的に行ってくれるツールが必要なのです。これはmakeそのものですね。

ウィンドウシステム上でのmakeのイメージとしてはmakeファイルを表示しているウィンドウがあってそれを実行することで処理が始まるというのがもっとも原始的な形態でしょう（図3）。このmake処理もマルチタスクで動いてほしいものです。ウィンドウシステム用にマルチタスクで動作するコンパイラ，アセンブラ，リンカなどのツール類がそろってくれば，makeを実行しながら，ほかのアプリケーションを動作させ続けるという芸当はたやすいことに違いありません。

ないものねだりを続けていますが，結局のところは，SX-WINDOWのシステムを抜けないでプログラムを作れることが最低条件です。

これだけを満たせばよいのであれば，現時点でも方法がないこともありません。フリーウェアのSXconsoleを使用すればよいのです。これは標準入出力をSX-WINDOW上でエミュレートすることを目的として作られたアプリケーションです。標準入出力のファイルへのリダイレクトは，まだサポートされていませんが（きっとそのうちにサポートされるでしょう），scanf関数やprintf関数を用いて作られたC言語のプログラムを実行すれば，SXconsoleのウィンドウ内から標準入力へのデータをキーボードから入力したり，標準出力へのデータをウィンドウ内に表示することができる

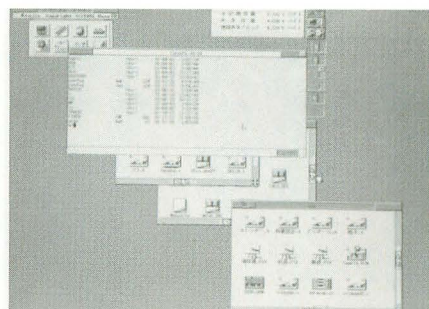
SXconsole

シェルを呼び出す

ようになっているのです。

このSXconsoleにはSHELLという機能があって，ウィンドウ内でHuman68kのコマンドを実行することができるようになっています。つまり，SXconsoleはシェル（コマンドインタプリタ）ウィンドウとしても使用できるわけです。この機能を利用すればSX-WINDOWから抜け出ることなしにコンパイルやリンクなどを行うことができます。

ただし，欠点もあります（とはいえ，これはしかたない）。それはSHELL機能を終了する（EXITをコマンド入力する）まで，ほかのアプリケーションの実行が停止してしまうことです。結局は，SX-WINDOWをいったん抜けてHuman68k上でなにか仕事をし，そのあとにSX-WINDOWを再起動するのと同じことです。しかし，起動時に次々とウィンドウをオープンし直す手間（結構長い時間がかかる）がなくなる分だけ，SXconsoleを使用するほうが有利なのです。なお，SXconsoleの作者はハイスピードリンカ（HLK.X）でお馴染みのそるとさんです。おそらく，THINK CのANSIライブラリのコンソール機能に影響を受けて作成したものなのでしょう。

とにかく，現在でもSX-WINDOWを抜け出すことなくコンパイルなりリンクを実行することは一応可能です。SX-WINDOWの開発環境でもっとも深刻な問題は，SX-WINDOW上で動くデバッガがないということでしょう。

ソースコードデバッガなどという贅沢は申しません。アセンブリ言語レベルでデバッグする方法さえほとんどないのが実情です。Macintoshの場合と同じく，SX-WINDOW用のプログラムを書くことは非常に難しいのです。バグの2匹，3匹は当たり前の世界です。SX-WINDOW用のプログラムを書いたことのある人なら誰でも，ちょっと変なプログラムを書いたために，バスエラーとかシステムエラー（なんだこれは）が発生してSX-WINDOWのシステムをハングアップさせた経験が一度ならずあると思います。

現在，SX-WINDOWのプログラムをデバッグする方法は，Human68k上で動作する（SX-WINDOW用に改造された）シンボリックデバッガのDB.Xと，SXWDB.Xというデバッグ用のSXシェルを組み合わせて使うことによって行うことができます。本来は，シリアル回線を通じて，別のターミナルとの間でコマンド入力やメッセージ表示を行うことになっているようです。しかし，出力で画面が乱れるのを気にしなければ，ターミナルは必要ありません。

ただし，Oh!Xの付録ディスクで配布され

図3

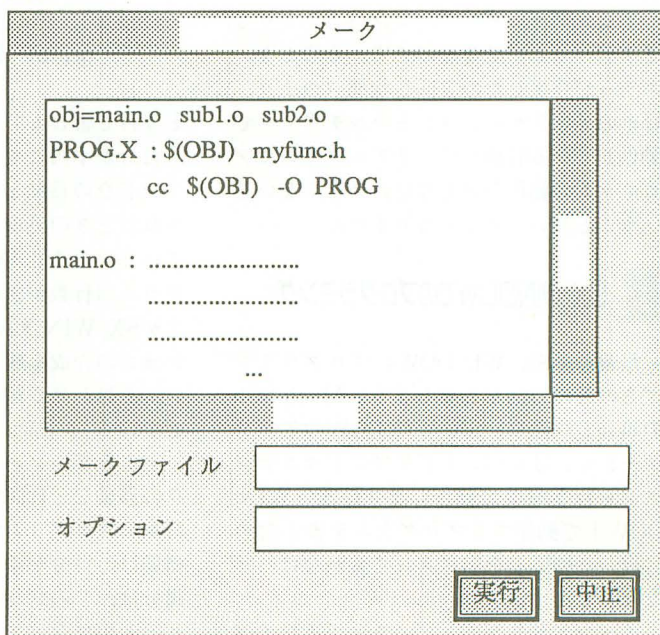
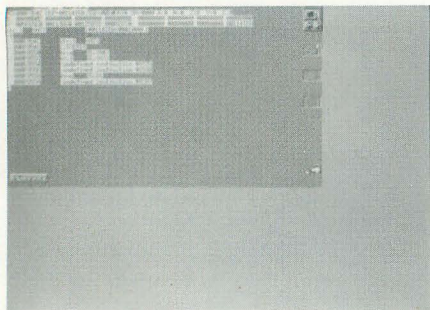
メーク

```
obj=main.o  sub1.o  sub2.o
PROG.X  : $(OBJ)  myfunc.h
          cc  $(OBJ)  -O PROG

main.o  : ........................
          ........................
          ........................
          ...
```

メークファイル

オプション

実行　中止

デバッグのようす

たsxlib.aをリンクして作ったプログラムでは，Human68kのコマンドラインから起動されたとき，SXシェルとしてSXWDB.XではなくSXKERNEL.Xを起動するようになっています。したがって，リネームするかライブラリを書き換えない限り，それらのプログラムの動作をデバッガで追うことはできません。

また，一部のアプリケーションプログラムでは，非公開の$A350というシステムコールでSXシェルを呼び出しています。このようなプログラムもデバッガで追うことはできないでしょう。デバッガでプログラムを実行できるような場合でも，最初にHuman68k上からDB.Xを立ち上げなければならないのが嫌なところです。

やはり，SX-WINDOW上で動作するシンボリックデバッガなりソースコードデバッガがほしいところです。機能的には，Human68k上のDB.XやSCD.Xで十分と思います（インタフェイスは変える必要があるかもしれません）から，これがSX-WINDOWに移植されることを期待しましょう。

アセンブラ，リンカ，コンパイラよりも，まずデバッガが必要なのです。そのとき，エラーが発生したのと同じ状態（つまり，ほかのアプリケーションとマルチタスクで動作している状態）で，リアルタイムにプログラムの動作を追えなければならないということはいうまでもありません。

## SX-WINDOWでのプログラミング

最後に，SX-WINDOWのプログラミングそのものの話をしましょう。Macintoshの場合と同様に，SX-WINDOWのアプリケーションはイベントドリブンでマルチタスクを行う環境で動作します。SX-WINDOW上で動作するプログラムを書くためには，このことを頭に入れておかなければなりません。

イベントドリブンとは，さまざまなイベント（事象）がきっかけとなってプログラムが駆動されるということです。ウィンドウシステムでは対話性のよさを第一とします。ユーザーがどのような操作をしようが，それ（によって生じるイベント）に対して，適切な処理を行えるようにプログラムを書かなければなりません。

プログラムの構造としては，このイベントに対してはこの処理，あのイベントに対してはあの処理……，というように，発生するイベントとその処理を並べたものになります。また，マルチタスクを行わなければならないという点では，切りのいいところで自分の処理を（なるべく早く）中断して，制御をシステムに返すということを考慮しなければなりません。

イベントドリブンというプログラム構造は対話型のプログラムを書いたことがある人ならば容易に理解できると思います。しかし，やっかいなのは，プログラムの本来の動作と直接関係のない「ウィンドウの移動」とか「ウィンドウの拡大」とかいったイベントまでがアプリケーションに通知されてくることです。それらのイベントに対しても，対応する処理をいちいちプログラムの中に書かなければならないのです。このため，ちょっとしたことを行うプログラムでもソースコードの行数は大変なものになります。たとえば，ウィンドウ上に，

Hello, world.

と表示するだけのプログラムでも，100行程度のコード量が必要になってしまうのです。Humna68k上で，画面に同じメッセージを表示するだけなら，C言語で，

```
main()
{
    printf("Hello, world.¥n");
}
```

と4行で書けるプログラムが，です。

自分が本当にやりたいことのほかに，ウィンドウの移動とか，マウスのボタンが押されたときの処理とか，関係ない（と思っている）たくさんのイベントの処理でプログラムの行数が膨れあがるのです。このことがSX-WINDOWで動作するアプリケーションの作成を難しい（あるいは，面倒臭い）と思わせる最大の要因になっているのです。

ところで，ソースコードの中でウィンドウの移動などに関する処理を行う部分は，ほかのプログラムでもほとんど同じです。結局は，自分が本当にやりたいこと以外の部分は，どのプログラムでも似たり寄ったりのものになってしまいます。

そこで，多くの場合，ウィンドウの移動などの定型的な処理があらかじめ記述されているスケルトンという骨格プログラムをコピーしてきて，自分がやりたい処理を書き加えるというプログラミングスタイルになってしまいます。しかし，スケルトンがあるからプログラミングは簡単なんだよ，という意見は本末転倒のような気がします。プログラムの側で定型的に書ける処理ならば，それをシステムの側で処理してくれてもよさそうなものです。

実際，

ウィンドウの移動
ウィンドウのサイズ変更
アイコン化
マウスカーソルの形状変更

などの処理はシステムの側だけで十分対応可能だと思います。あるいは，ウィンドウ操作に関するこれらの処理をするプログラムが差し替え可能な常駐タスクとして存在するのも面白いでしょう。

プログラムの在り方としては，定型的に書き下せるような処理に関してはなにも考えず（マルチタスクは考えなければならないが），本来自分の行いたい処理だけに専念できるのがもっとも望ましいのです。このような状態になれば，SX-WINDOW上で動作するプログラムの作成は，いまよりもずっと楽になるに違いありません。

ただ，その場合でも，システムの側ではなにをしていいかわからないウィンドウのアップデート（書き換え）処理だけは，プログラムの側で負担するのもやむを得ないでしょう。

## これからのSX-WINDOW

コンパイラやmakeなどの統合開発環境の充実，複雑なプログラムからの解放など，これからのSX-WINDOWに望むことは非常にたくさんあります。しかし，各種のフリーソフトがネットで発表され続けているだけでなく，開発元のシャープからもエディタやEasyPaintといったアプリケーションが発表がされるなど，ゆっくりながらも着実な進歩を遂げてきているSX-WINDOWですから，どのような希望（あるいは夢）もいつかは実現されると信じています。

さて，話は変わりますが，現実的な問題としては，早くリソースエディタを発表してほしいと思います。それが不可能ならばリソースファイルの形式だけでも公開してくれないものでしょうか（自作しようにも作れない）。

ウィンドウプログラムへの招待

# SXアプリケーションの基本構造

Ishigami Tatsuya 石上 達也

「難しい」というイメージの強いウィンドウアプリケーションのプログラミング。面倒な処理が多いのですが基本的な部分を用意すれば意外と単純なものです。付録として必要最低限のSXコールもまとめてみました。

「使えば天国,作れば地獄」とか,「使う人：Window＝「窓」,作る人：Window＝「壁」」とかいわれるWINDOWシステムですが,Human上でのプログラミングに比べて,いったいどこがどのように「地獄」であり,「壁」であるのでしょうか？

白鳥は華麗な泳ぎを見せても，その水面下では大変な努力をしているそうです。その努力を知ってこそ，本来の白鳥の美しさを知ることができるのです，と，知ったかぶって，今回は編集部のなかでも比較的SX初心者である私が,SXプログラミングの初心者と同じ視点から泳いでいる白鳥をひっくり返してSX-WINDOW上アプリケーションの観賞法を探っていきたいと思います。

なお，この記事では，なるべく概念的な話をするつもりなので，具体例がほしい方は，中森氏の連載記事や付録ディスクのプログラムをのぞいてみてください。また，一部，適切とはいえないような表現がありますが，入門記事ということで見逃してください。

注）ここでいう初心者とは，最低限C言語の基礎を理解している「SX-WINDOWプログラミングの」初心者のことです。

## C言語の復習と環境作り

現在，SX-WINDOW上のアプリケーションを作成するには，C言語を使うかマシン語でプログラミングを行うかの2種類があります。ほかにもあるにはあるのですが,一般性がないのでここでは無視します。

また,私がMC68000のマシン語をよく知らないのと，マシン語で説明をしだすとややこしくなり，結局は「わかる人にはわかる」ということにもなりかねないので，ここではC言語を用いて説明します。C言語について説明する機会ではないので，読者の皆さんはある程度C言語について知っているものとして話を進めさせてもらいます。

もっともC言語のエキスパートである必要はなく，とりあえず，構造体，インクルードファイル，auto変数とstatic変数の区別，そして，とおり一遍の文法がわかっていれば十分です。

逆に，これらの知識とある程度のプログラミングセンスがないとSXアプリの作成はつらいものがあります。そうそう，気持ちの準備のほかにも，ハードディスクや4Mバイト以上のメモリ，そして，1991年1月号の特別付録「謹賀新年PRO-68K」が必要となります。

## イベントドリブン

Oh!Xにはあまり見かけませんが,DOS関係の雑誌では，よくグラフィック描写とか,音源ドライバなどの関数パッケージの広告を見かけます。どうせ，そのようなシステムよりのプログラムは誰が書いても同じなのだから，そのプログラムを外部から買ってきて，プログラマはもっと創造的な仕事をしよう，という発想なのでしょう。

マウスのボタンが押されたのを感知するとか，ウィンドウの大きさを変更するなどのプログラムは誰が書いても，だいたい同じようなサブルーチンになります。そこで,SX-WINDOWでも，このような単調なサブルーチンはあらかじめシステム側に組み込んでしまい，アプリケーションプログラマは「マウスのボタンが押された」あとのことのみを考えればよいようになっています。つまり，システムに対してなにか出来事があった場合，その感知はシステムが行い「そのときどうするか」のみをアプリケーションプログラマが考えればよいようになっています。

話は変わりますが，ごく正直に設計されたマルチタスクシステム上で,「15パズル」とその他のウィンドウが同時に開かれている場合を考えてみてください。たぶん，ほとんどの時間を「15パズル」は，自分のウィンドウに対して，キーボードが押されるのを，まだかまだかと首を長くして待っていることでしょう。

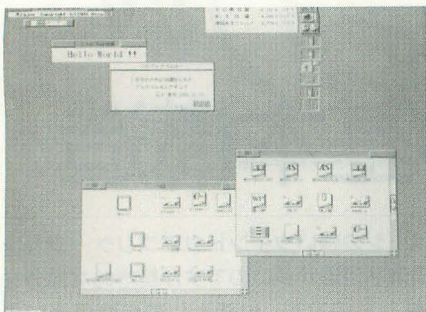
つまり，MPUはタスクが「15パズル」に切り替えられると，ほとんどの時間をキーボードの入力待ちループで費していることになります。もし，あなたが留守中の友達に電話をかけてしまったとき，1分おきに電話をかけるでしょうか？　たぶん，留守番電話などに，帰ったらこちらに電話をくれるように吹き込んでおくでしょう。これと同じように,SX-WINDOWでも,タスクが毎回キーボードが押されているかを（積極的に）調べるのではなく，キーボードが押されたらタスクが（消極的に）呼び出されるという方法を採用しています。こうすれば，キーボードの入力待ちループで時間を浪費するということが解除できます。

キーボードが押されるなどのように，システムにとってなんらかのきっかけになるような出来事をイベントと呼び，イベントをきっかけとしてシステムを駆動するやり方をイベントドリブン方式といいます。このイベントドリブン方式によって，SX-WINDOWではプログラミングの容易性,時間の有効活用を実現しています。

## こんにちは世界！

まず，C言語でいちばん初めに書くプログラムといえば（実際にやったかどうかは別として）通例的に「画面に“Hello World!!”と表示する」プログラムでしょう。これは，以下のようにC言語を知っていれば誰でも組めるほど簡単なプログラムです。

```
main( ) {
 printf("Hello World !!¥n");
}
```

さて，SX-WINDOWで同じことをやろうとすると，ちょっと事態が変わってきます。Oh!X1991年2月号での村田敏幸氏によるプログラムを参考に概算すると（氏のプ

こんにちは世界！

ログラムの143行～230行までを除くと)，ざっと230行に及びます。もちろん，コメントや空白行も含まれているので，その分を取り除いて考えなければなりませんが，どう低く見積もってもこの勝負，3行対100行ぐらいにはなってしまいそうです。この差はいったいどこから生じてくるのでしょうか？

SX-WINDOW上のプログラムには，「独自のウィンドウを開いて，その中で動作を行う」という決まりがあります。当然，そのウィンドウはSXの流儀に従っていなければならず，ウィンドウの移動やリサイズ，そして場合によってはアイコン化やコントロールメニューもサポートしていなければなりません。

これに対し，Human上でのプログラムはこんなことはできません。つまり，画面に文字列を表示することだけを考えてやればよいのです。

ただの「画面に“Hello World !!”と表示する」プログラムでさえ，こんなにも(付加的な）機能に差があるのです。そして，その差をどういう差としてウィンドウ上で実現するかという情報は（実現しないというのもひとつの情報ですが)，やっぱりプログラマがコンピュータに教えてやらねばなりません。結局，この情報量の差が，3行対100行になってしまうのです。

しかし，3行対100行といっても，それほど恐れることはありません。どのプログラムもたいていは，左のボタンが押されたら

リスト1

```
drawWindow()
{
    point_t    loc;

    loc.p.x = 40;
    loc.p.y = 6;
    GMSetGraph((graph*)winPtr);
    GMPenMode(G_BACK|G_PSET);
    GMFillRect(&updRect);
    GMPenMode(G_FORE|G_PSET);
    GMFontKind(G_ROM24);
    GMMove(loc);
    GMDrawStrZ("Hello World !! ");
}
```

移動もしくはリサイズで，右のボタンが押されたらコントロールメニューを表示する，というようにSXの流儀に沿っています。どのプログラムでも同じような機能になりますので，一度作ってしまえば，あとはその関数を使いまわしてしまえばよいのです。

幸いなことに，本誌の読者には中森章氏による「良い子のためのSX-WINDOW入門」という連載記事があります（最近は，あまり見てないなぁ……)。この記事中では，使い回せるような関数の集大成として，非常に完成度の高いスケルトン（骨格）プログラムが掲載されています（中森さん，今度はテキストマンの解説よろしくお願いします。へこへこ)。

## 「良い子のSX-Window入門」利用法

以上は，なるべく機能的に近いサンプルをということで，1991年2月号の記事との比較を行いました。しかし，スケルトンとして用いるときの完成度や，バックナンバーの在庫の関係から，ここでは中森 章氏の連載記事「良い子のためのSX-Window入門」で紹介されているスケルトンプログラムの利用法を考えてみましょう。皆さん，お手元に1991年10月号の準備はよろしいでしょうか。今回は，リスト1（10月号の）のみで結構です。

このプログラムは，完全なスケルトンプログラムではなく，マウスカーソルを変更するためのサブルーチンが随所に埋め込まれています。まず，これらの削除を行います。

●今回はマウスカーソルを変更するようなことは行わないので，構造体TXcsr関係の宣言はカットしておきます。77行～94行を削除。

●左のマウスボタンでの機能には，クローズしか持たせないので，その他の処理は入りません。また，グローボタンが操作されるようなこともないのですが，スケルトンということでここの部分は取っておきましょう。367行～386行を削除。

以上のような変更で完全なスケルトンプログラムになりました。このプログラムは，今後，SXアプリケーションを作成しようとする際に，土台となるプログラムですから，この状態のまま1本取っておいたほうがよいでしょう。

次に，ウィンドウサイズやタイトルなどの変更を行います。

●アイコン時の横サイズを250に変更

19 #define ICON_WIDTH 250

●ウィンドウサイズを横250，縦30に変更

25：#define WINWIDTH 250

26：#define WINHIGHT 30

●タイトルを変更しておきましょう。タイトルなんてなんでもいいのですが，“こんにちは世界”とでもしておきましょう（我ながら情けないタイトル)。

27：#define WINTITLE “¥016こんにちは世界”

●タイトルを変更すると，自動的にメニューも変更しなければならなくなります。

32：#define MNLIST “¥0¥0¥027こんにちは世界について ¥0¥0¥013アイコン…”

●ついでに，ダイアログメッセージも変更しておきましょう。

562：“¥001¥026こんにちは世界を表示する”

いったいなんのメッセージかわからない人は，1991年10月号を見てください。

●このプログラムではコントロールスイッチを使わないので，以下の変更を行います。

240：ctrlFlag = FALSE;

最後に，画面に文字列を表示するためのサブルーチンを追加して完成です。

●元のプログラム172行～209行の関数drawWindowを削除。関数drawWindow（文字どおり，ウィンドウ画面内に描写を行う関数）が，今回の主役です。当然，今回の目的に合わせて作り直します。リスト1に示す関数に替えてください。

●このプログラムは，ただ，ウィンドウの中に文字列を表示するだけのものです。よって，アイドルイベント（サイン会とか，握手会などではなくて，システムになにも起こっていないときのイベント）時にはやることがありません。311行～341行を以下のように変更します。

```
procIDLE( ) {
    return( FLASE );
}
```

これで，「画面に“Hello World !!”と表示する」プログラムができました。どうです？ スケルトンプログラムがあれば，簡単でしょ。

## 注意点

軽いノリで説明してきましたが，私も，あのドキュメントをあっさりと解読できたわけではありません。いくつか，SXアプリケーションを作成するうえでつまずいた概念が出てきましたので，簡単に紹介しておきます。

●メモリハンドル

SX-WINDOWは，いうまでもなくマルチタスクウィンドウシステムです。複数のタスクと呼ばれるプログラムが，たった1個のMPUと一連のメモリ空間とを使って動作しています。これらのタスクの中には多量のメモリエリアを必要とするものもあり，それが必要になると，SXのシステムのなかでも特にメモリ関係の処理を担当している「メモリマン」におうかがいを立てて自分の使ってよいメモリ領域を確保してもらいます。そして，不必要になると「メモリマン」に「ありがとね」といって，メモリ領域を破棄してもらいます（つまり，ほかのタスクで使えるようにしてもらう）。

さて，このような動作をウィンドウを開いたり閉じたりするごとにやっていると，いつの間にかメモリの使用状況は図1.1のようにぎたぎたになってしまいます。

SX-WINDOW上で「プロセス情報」をドラッグすると，「メモリ空き容量」と「メモリ連続空き容量」と2種類の「メモリ空き容量」が報告されますね。これは，前者が純粋に使っていないメモリブロックの総和を，後者が使っていないメモリブロックのなかでも最大のものを報告しているのです。つまり，両者の差が図1.1にあるような「メモリ領域の隙間」になるのです。

そして，メモリ空間が隙間だらけになると，「メモリ連続空き容量」が減少してしまいます。やがて，減少しすぎて，ある大きさの連続したメモリブロックが確保できなくなってしまいます。ハード的にメモリが不足しているのであればあきらめもつきますが，メモリブロックの確保のしかたがちょっと悪かったために，確保できるはずのメモリブロックが確保できないというのはもったいない話です。

そこで，考えられたのがメモリコンパクションと呼ばれる機能です。これは，メモリの隙間がある程度たまってくると，図1.2のように「えいやっ！」とメモリブロックを詰めて「メモリ連続空き容量」を目一杯確保できるようにする機能です。

このコンパクションの際，メモリブロックのなかには，「どーしても移動させられちゃ駄目」とだだをこねる奴や，逆に「そんなに困っているのなら，もう俺は消えてやるよ」という寛大な奴もいるのですが，これらは例外的な存在ですので，ここではタスクが確保したメモリブロックはコンパクションによって移動するものとして考えます。

また，このコンパクションは積極的に行う必要はありません。SXのシステムが，暇を見つけては勝手に行ってくれます。どういうときにシステムが暇かというのは，付録ディスクのリファレンス中，各マニュアルの関数の説明で「コンパクションが発生する可能性があります」と書かれているか否かでわかります。

と，このようにSX-WINDOWでは，メモリブロックの位置がことあるごとに移動してしまいます。そこで，とあるメモリを指示する機能として，ポインタの代わりにメモリに収められた情報を管理する方法としてメモリハンドルという概念が使用されています。

私（メモリの内容の例だと思ってください）の住所（アドレス）は都内某所ですが，よく引っ越し（コンパクション）をします。で，せっかく読者の皆さんが「石上達也に励ましのお便りを書こう！」（笑）と思っても，その住所（実効アドレス）に私がいるとは限りませので，宛先がわかりません。

ところが，私は引っ越すたびに編集部（メモリハンドルに相当）に報告していますので，手紙を出す直前に（メモリアクセスする直前に），編集部に私の住所を問い合わせてからなら手紙を投函できるのです（編集注：筆者の個人的な情報は公開していません）。

C言語でいうと，

```
char    *ptr;
  strcpy("Hello World !!¥n", ptr);
```

は，メモリポインタptrで示されるメモリ空間に，直接，文字列を転送し，

```
char    **Hdl;
 strcpy("Hello World !!¥n",*Hdl);
```

は，メモリハンドルHdlにまずおうかがいを立て，そして文字列を転送すべきメモリ空間のアドレスを教えてもらい，そのメモリ空間に文字列の転送を行います。両方の変数の宣言の方法を見比べてください。後者の宣言は，

```
char    *(*Hdl);
```

と同じことですよね。実はハンドルというのは「ポインタへのポインタ」だったのです。

●リエントラント

普通に考えると，ひとつのウィンドウにひとつのプログラムあり，なのですが，特殊な場合として，同一のタスクが複数個開かれていた場合を考えましょう。もちろん，ひとつのウィンドウごとに，ひとつのプログラムを起動してもよいのですが，これらのプログラムコードはまったく同じものです。ただ，プログラムの配置されているアドレスが異なるだけです。

これは，メモリ状況の緊迫しているX68000においては，もったいないお化け発生注意報ものの事態です。できることならば，同じプログラムで複数のタスクに対応させたいものです。

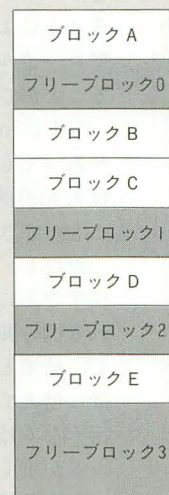
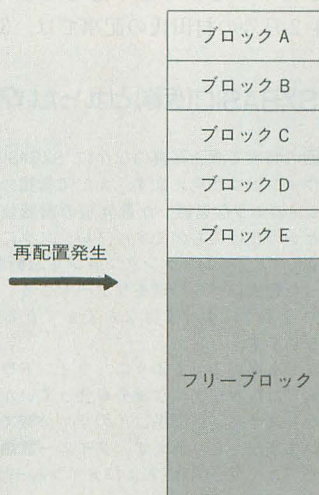
この条件をなんとかクリアできたものをOBJR型プログラムと呼び，残念ながらクリアできなかったものをOBJC型プログラムと呼びます。さらに軟弱なことに（というか，諸々の事情により），一度に複数実行できないプログラムをOBJO型プログラムと呼びます。軟弱な例ではありませんが，日本語入力を行うhenwin.xは複数同時に実行させても意味がないので，OBJO型のプログラムです。

さて，以上のようにプログラムの分類があるのですが，SXの作法としては，特別な事態が生じない限りプログラムはOBJR型にすべし，ということになっています。C言語でこのようなプログラムを作成するとき，いちばんの問題点は「グローバル変数が使えない」，このひと言に尽きるでしょう。

たとえば，

```
int   cnt;
main( ) {

for(cnt = 0; cnt <= 10; cnt++)
```

図1

```
    printf("cnt=&d¥n",cnt);
  }
```

というプログラムを，なんとかうまくアレンジしてSX上に載せたとしましょう。ところが，このタスクを複数一度に走らせると以下のような現象が生じてしまいます。

1) タスク1でとりあえず，0から2までを表示する。

2) ここで，タスクの切り替えが起こりタスク2が起動される。

3) タスク2では，変数cntに0を代入し0から1までを表示する（なにかの拍子で，タスクごとに割り当てられる作業量が替わってしまうのは日常茶飯事）。

4) ここで，タスクの切り替えが起こりタスク1が起動される。

5) タスク1では3から表示をしなければいけないのだが，タスク2の影響があって2から表示を開始してしまう。

これは，タスク1，2で変数cntの衝突が起こってしまったためです。

これの解決法として，

1) グローバル変数を使わない。

2) OBJR型のプログラムにするのをあきらめてしまう。

3) グローバル変数を構造体としてひとまとめにし，その構造体のアドレスを各関数で共用する。

などがあります。

1)の方法では，プログラムの種類や構造が極端に制限されてしまいます。あまり，実用的とはいいがたいでしょう。「良い子のためのSX-Window入門」や付録ディスクに入っていたサンプルディスクでは，2)の方法を使っています。これは，説明のためにできるだけプログラムをシンプルにしなければならないためで，実用プログラムでこれを行うことはあまり感心しません。1991年2月号の村田氏の記事では，3)の方法を用いてました。これは言葉で説明するとかえってややこしくなるので，プログラム中からエッセンスだけを取り出して，復習していくことにしましょう。

まず，プログラムの冒頭で

```
  typedef struct {
    window   *winptr;
    BOOLEAN activeflag;
    rect   winsize;
    tsevent eventrec;
  } GVAL;
```

と，本来ならグローバル変数にしてしまうべき変数をひとまとめにして，GVALという構造体を作成します。

ここで，windowとかrect,tseventなどという宣言が出てきますが，これはSX-WINDOWのプログラムを作るために必要な変数の型です。付録ディスクに収録されていたインクルードファイルsxdef.hの中で定義されていますので，中身が気になる人はのぞいてみてください。そうでない人は，次へ行きましょう。

main関数の始まりでは，

```
    GVAL    gval;
```

と宣言しているのがわかります。これならば，GVAL型の構造体変数gvalは，main関数が呼び出されるごとに（それぞれ別の位置の），メモリ上に確保されますので，リエントラントなOBJR型のプログラムが書けそうです。次に，

```
  if(!init(&gval))
    term(&gval, EXI_TFAILURE);
```

と，プログラムの初期化を行い（init( )関数），もしそれに失敗したらプログラムを終了します(term( )関数)。呼び出される側の関数，たとえばinit( )を見てみると，

```
  init(GVAL *gp) {
```

と，パラメータを宣言しています。こうすることによって，main( )で確保されたauto型ローカル変数gvalを関数initでも共有することができるようになります。

もし，initから呼び出される孫受け関数があっても，同じような関数コールを行えば大元の関数mainと同じ変数を共有することができるようになります。

ただし，この方法を使うと，関数を呼び出すごとに，変数gpのアドレスを呼び出される側の関数に教えなければなりませんので，結果としてパラメータが常にひとつ余計に増えてしまいます。これは，見た目が汚くなるほかにも関数コールのオーバーヘッドが重くなるという欠点を持っています。

以上はXCを用いてプログラムを開発するときのお話でしたが，SXのシステムでは，同じタスクには，a5レジスタに同一の値を保存しておくという性質がありますので，gccを持っている場合には，

```
  register GVAL *gp asm("a5");
      /* 大域変数モドキへのポインタ */
```

とでも冒頭で宣言すれば，あとはグローバル変数（モドキ）へのアクセスは変数名の頭に"gp->"と4文字足せばよいだけになります。

```
例) gp->activeflag = YES;
```

gccをもってすれば，リエントラントなんてこんな簡単に実現できてしまいますので，皆さんもなるべくOBJR型のプログラムを作成するようにしましょう。

## 最後に

「ジーザスII」というゲームがあります。なんでも原作はグラフィック画面を640×200，8色の解像度しか持たないPC-8801で作ったそうです。多くの人たちは，ちょっとねぇ，と感じたようですが，私は逆によくやってくれたと思いました。水墨画家に向かって「世の中には絵の具というものがあるのだから，それを使ってカラーの絵を描きなさい」という人がいるでしょうか。同様に，「ジーザスII」は，自ら640×200，8色という限られた世界を作り上げて，その中で勝負をかけたのでしょう。

創造的な活動を行おうとするとき，必ずなんらかの制限にぶつかります。この制限事項をそっくり回避してしまうか，それとも正面から取り組み，それをテーマとして捉えるかは人にもよるでしょう。ただし，後者には前者にはわからないような一歩上の視点から，物事をながめることができるようになります。

同様に，SX-WINDOW上でプログラミングをするためには，かなりの数の制限事項があります。では，これらの制限事項を克服するためにはなにをしたらよいでしょうか？ とりあえず，なにかプログラムを作ってみてください。最初のうちは簡単なものから始めます。それこそ，画面に"Hello World !!"と表示するプログラムでもよいでしょう。そして，なにか問題点に当たったら，他人の作ったプログラムを見てみましょう。その作者もきっと同じような問題点にぶつかっているはずです。その解決方法をプログラム中から，見事抜き出せれば，しめたものです。このような作業を地道に繰り返せば，きっと光は見えてきます。光が見えたら……，「Oh!X」はいつでもあなたの投稿を待っています。

### SXBASIC(仮称)とはいったい?

今回の特集で画面写真のなかに"SXBASIC"というウィンドウが見えます。まだ「C言語のようなPascalのような言語」が最低限の言語仕様で動くというだけのものですが，「レジスタに値を入れる」，「Aラインファンクションを起動する」といった超低レベル処理をサポートしていますのでやりようによってはなんでもできるはず……なのです。

本文でも書いているように，ウィンドウプログラムでは誰かが面倒な部分を作っていかなければなりません。処理系はそのうち公開できると思いますが，とりあえず，タイニー言語ということでユーザー関数(およびメインルーチン)の充実に期待しましょう。

# よく使うSXコール一覧

## コントロールマン(Control Man)

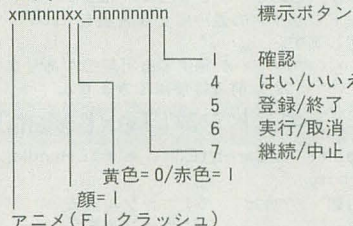
●control ＊＊CMOpen(window ＊Window, rect ＊bounds, LASCII ＊title, int visible, int value, int min, int max, int ID, long user);
返り値　0のときはエラーが発生しています。errno のエラーコードを参照してください。
コール　$A 289
機能　コントロールを作成します。
説明　Window はコントロールを書くウィンドウを指定します。
bounds はコントロールの大きさと位置を指定します。Window上のローカル座標で指定します。
title はコントロールのタイトルを指定します。コントロールによっては無視されます。
visible は描画許可スイッチで，真（0以外）で許可します。ただし visibleが描画許可状態でも実際には書かれません。CMDraw() / CMDrawOne() を実行してください。
value は－32768～32767の値を取ります。コントロールの初期値です。
minはWORDでvalueの最小値の初期値です。
maxはWORDでvalueの最大値の初期値です。
IDはresourceIDです。16倍してください。
userはcUserの初期値です。
<compact>が発生します。

resource ID
0　標準ボタン
1　セレクトボタン
2　オルターネイトボタン
19　スライドボリューム
20　アップダウンボタン
64　スクロールバー（横，ウィンドウ用）
65　スクロールバー（縦，ウィンドウ用）
66　スクロールバー（横）
67　スクロールバー（縦）

●int CMDispose(control ＊＊Control);
返り値　0のときはエラーが発生しています。errno にエラーコードを参照してください。
コール　$A 28 A
機能　指定のコントロールを削除します。
説明　このコントロールに関連したすべてのメモリを解放します。さらにコントロールリストから外し，ウィンドウ上に表示されているなら消し去ります。
<compact> が発生します。
● int CMValueGet(control ＊＊Control);
返り値　control value が返ります。value は下位WORD に入ります。したがって int の負数が入っているときはエラーコードを示します。
コール　$A 291
機能　指定のコントロールの値を読み出します。
● int CMDraw(window ＊Window);
返り値　終了コードが返ります。終了コードが負の場合はエラーです。
コール　$A 28 E
機能　指定のウィンドウに付属するすべてのコントロールを描画します。
説明　不可視属性のものは描画しません。CMShow() を使ってください。
<compact> が発生します。

## ダイアログマン(Dialog Man)

●dialog ＊DMOpen(dialog ＊Dialog, rect ＊bounds, LASCII ＊title, int visble, int wDefID, window ＊behind, int cBox, long taskID, dlglList ＊＊dItems);
返り値　0のときはエラーが発生しています。errno にエラーコードを参照してください。
コール　$A 2 C 3
機能　ダイアログウィンドウを作成します。
説明　Dialog が0のときメモリマンを使ってメモリを確保します。
ローカル変数としてスタックなどにデータ構造を確保した場合は，そのアドレスを渡してください。
bounds, title, visible, wDefID, behind, cBox, taskID は WMOpen()とまったく同じものです。そのまま WMOpen() に渡されます。
dItems は dlglList への handle です。
<compact> が発生します。
●int DMDispose(dialog ＊Dialog);
返り値　終了コードが返ります。終了コードが負の場合はエラーです。
コール　$A 2 C 6
機能　ダイアログウィンドウを消去します。
説明　DMOpen(DMRefer) で確保したメモリを解放し，指定の dialog をウィンドウリストから削除します。もし画面に表示されているなら画面より消去します。
DMOpen(DMRefer) で Dialog に0を指定したときに使用してください。
<compact> が発生します。
●int DMControl(void (＊proc)());
返り値　アイテム番号が返ります。終了コードが負の場合はエラーです。
コール　$A 2 C 7
機能　ダイアログのアイテムの受け答えをします。
説明　帰還属性を持つアイテムが押されるまでアイテムの制御をします。
proc は DMControl() が受け取ったイベントを変更または無視させるためのフィルタへの pointer です。フィルタ関数filter(dialog ＊Dialog, event ＊ev)内ではev を変更します。
<compact> が発生します。
●int DMBeep(int count);
返り値　負の場合はエラーです。
コール　$A 2 D 7
説明　beep 音を count－1 回鳴らします。

●int DMError(int mode, char ＊String);
返り値　アイテム番号が返ります。ボタン番号は左側のボタンから1，2です。
終了コードが負の場合はエラーです。
コール　$A 2 F 6
機能　簡易エラーメッセージ用ダイアログを開きます。
説明　mode は以下のようなビットフラグになります。

String は警告またはエラーメッセージを指定してください。また，あまり1行が長すぎると表示しきれませんので $0d を適当に入れてください。
<compact> が発生します。

## イベントマン(Event Man)

●unsigned long EMDClickGet(void);
返り値　ダブルクリック基準時間が返ります。
コール　$A 0 B 0
機能　ダブルクリックの基準時間を読み出します。
説明　ダブルクリックの判定はアプリーケーションが行います。
●int EM_Still(void);
コール　$A 0 AA
機能　マウスの左ボタンが押されたままになっていれば真を返します。
●int EMMSLoc(void);
返り値　上位 WORD に水平，下位 WORD に垂直方向の座標が返ります。
コール　$A 0 A 7
機能　マウスの座標を返します。ただしカレントポートによるローカル座標です。
●unsigned long EMSysTime(void);
コール　$A 0 AF
機能　現在のシステム時間を返します。単位は1/100秒です。

## グラフマン(Graph Man)

●int GMOpenGraph(int scrKind, graph ＊graphPtr);
返り値　エラーコードが返ります。
コール　$A 12 D
機能　<graph> を新しく設定します。
説明　指定された <graph> を指定の <screen type> で初期化し，<current graph>にセットします。<bitmap> のためのメモリが <pointer> として確保されます。
内部処理で GMInitGraph(), GMInitBitmap()が呼び出されます。
必要なメモリが確保できない場合，エラーとなります。
scrKind の値は以下のとおりです。
0　<text type>
1　<graphic type>
graphPtrは疑似<pointer>が使用可能です。
<compact> が発生します。
●int GMCloseGraph(graph ＊graphPtr);
返り値　エラーコードが返ります。
コール　$A 12 E
機能　<graph> が使用しているメモリを解放します。
説明　指定された <graph> の中の <bitmap>, <visible region>, <clip region> などの占めるメモリを解放します。もし <graph> 自体が <pointer> などでメモリに確保されていた場合，自分で解放処理を行う必要があります。
graphPtrは疑似 <pointer>が使用可能です。
●int GMSetGraph(graph ＊graphPtr);
返り値　エラーコードが返ります。
コール　$A 131
機能　<current graph>をセットします。
説明　指定された<graph>を<current graph>にセットします。
graphPtrは疑似<pointer> が使用可能です。
●int GMAPage(int aPage);
返り値　それまでの<access page>が返ります。
コール　$A 149
機能　<current bitmap> の <access page> をセットする
説明　<current graph> 中の <bitmap> の <access page> を変更します。
<current bitmap> は <text type> でなくてはいけません。
<gScript> への記録を行います。
<screen type> の確認は行ってません。
●int GMDrawStrZ(char ＊strZPtr);
返り値　エラーコードが返ります。
コール　$A 192
機能　<ASCIIZ> 文字列を描画します。
説明　<current graph> の <pen location> の位置から <ASCIIZ> 文字列を描画します。
<font kind>, <font face>, <font mode>, <font size>, <fore ground color>, <back ground color> が参照されます。
strZPtr は疑似 <pointer> が使用可能です。
<compact> が発生します。
<gScript> への記録を行います。
●int GMFontMode(int mode);
返り値　エラーコードが返ります。
コール　$A 18 D
機能　<font mode> をセットします。

説明　<current graph> の <font mode> を設定します。
<compact> が発生します。
<gScript> への記録を行います。
<font mode> の値を以下に示します。

| | |
|---|---|
| 0 | <pset> |
| 1 | <and> |
| 2 | <or> |
| 3 | <xor> |
| 4 | <npset> |
| 5 | <nand> |
| 6 | <nor> |
| 7 | <nxor> |

●int GMLocalToGlobal(point_t pt);
返り値　<global 座標> が返ります。
コール　$A 13 E
機能　<local 座標> から <global 座標> に変換します。
説明　与えられた <local 座標> を <current graph> 上の <global 座標> に変換します。
●int GMGlobalToLocal(point_t pt);
返り値　<local 座標> が返ります。
コール　$A 13 F
機能　<global 座標> から <local 座標> に変換します。
説明　与えられた <global 座標> を <current graph> 上の <local 座標> に変換します。
●void GMMove(point_t pt);
コール　$A 16 E
機能　<pen location> を変更します。
説明　指定した <point> に <pen location> を移動させます。
pt は <local 座標系> の <point> です。
●int GMPtInRect(rect *rectPtr, point_t pt);
返り値　結果の状態を示す値が返ります。
コール　$A 156
機能　<point> が <rectangle> 内かどうか調べます。
説明　指定された <point> が <rectangle> 内にあるかどうかを調べます。
rectPtrは疑似 <pointer> が使用可能です。
ここで <rectangle> 内とは以下のように定義されます。
pt を (x,y), <rectangle> を (lt,tp), (rt,bt) とする場合,

lt ≦ x < rt
tp ≦ y < bt

の式が両方とも成立するとき。

## メモリマン(Memory Man)

●handle MMChHdlNew(long lsize);
コール　$A 021
機能　カレント <heap> から <relocatable block> を割り付ける。
説明　MMHdlNew(_curHeap) を実行する。成功なら <handle> を返す。失敗ならNIL を返す。
<compact> を起こす。
●pointer MMChPtrNew(long lsize);
コール　$A 01 E
機能　カレント <heap> から <nonrelocatable block> を割り付ける。
説明　MMPtrNew(_curHeap, lsize) を実行する。成功なら <pointer> を返す。失敗なら NIL を返す。
<compact> を起こす。
●int MMHdlDispose(handle hdl);
コール　$A 038
機能　<handle> を解放する。
MMHdlDispose() は例外的に引数に NIL を認める。このとき MMHdlDispose() はなにもしない。
常に Success を返す。
●int MMPtrDispose(pointer ptr);
コール　$A 02 F
説明　<pointer> を解放する。
MMPtrDispose() は例外的に引数に NIL を認める。このとき MMPtrDispose() はなにもしない。
常に Success を返す。

## メニューマン(Menu Man)

●int MNSelect(menu **Menu, point_t pt);
返り値　選択されたメニューアイテム番号
エラーコード(負のとき)。
コール　$A 268
機能　メニューを表示し，マウスの右ボタンが離されるまでマウスを追いかけて，メニューアイテムをセレクトします。
説明　pt は右ボタンが押されたマウス座標です。
メニューアイテム番号は，mData の直後に書かれたものが 1 です。0 はなにも選択されなかったことを示します。

## タスクマン(Task Man)

●int SXCallWindM(window *windowPtr, tsevent *eventrecPtr);
コール　$A 3 A 2
機能　ウィンドウマンを呼び出す。
返り値　パートコードが返ります。
説明　eventrecPtr のイベントレコードで指定されたイベントにあった処理を，windowPtr で指定されたウィンドウレコードに対して行います。
サポートされる処理は，ウィンドウの移動，ウィンドウサイズの変更，ウィンドウのズーム，クリップのオン・オフです。
<compact> が発生する可能性があります。
●int SXCallCtrlM(window * windowPtr, tsevent *eventrecPtr, void **CtrlHandleV, void **CtrlHandleH,rect *dsizePtr, control ***CHdl);
返り値　パートコード/error code(Nflag:ON)
コール　$A 3 A 3
機能　コントロールマンを呼び出す。
説明　eventrecPtr のイベントレコードで指定されたイベントにあった処理を，windowPtr で指定されたウィンドウレコードのコントロールに対して行います。
スクロールバー以降は省略可能です。
<compact> が発生する可能性があります。
●int SXCallWindM 2(window *windowPtr, tsevent *eventrecPtr, rect *rectPtr);
返り値　ウィンドウパートコード。負数ならエラーコード。
コール　$A 41 F
機能　ウィンドウマンを呼び出す。
説明　SXCallWindM()と同様の処理を行いますが，このファンクションではグロー時のウィンドウサイズの最小値と最大値を rectPtr で指定します。
<compact> が発生する可能性があります。
ver 1.02 以前では使用できません。

## テキストマン(Text Man)

●int TMCaret(tEdit **tEHandle, int mode);
引数　mode　0：カーソル消去
1：カーソル表示
返り値　負数ならエラーコード。
コール　$A 311
機能　カーソルを表示する。
説明　tEHandle で指定された <tEdit> レコードのカーソルを表示または消去します。セレクト部分がある場合はなにもしません。ハイライト表示の変更は，TMActivate 2 ( ) TMDeactivate 2()により行ってください。
<compact> が発生する可能性があります。
●int TMDispose(tEdit **tEHandle);
コール　$A 312
機能　指定された <tEdit> レコード(編集テキストを含む)を廃棄する。
説明　tEHandle で指定された <tEdit> レコードを廃棄します。使用されているすべてのハンドルを廃棄します。
<compact> が発生する可能性があります。
表示はしません。
●int TMUpDate(tEdit **tEHandle, rect *rUpdatePtr);
返り値　負数ならエラーコード。
コール　$A 313
機能　指定された <tEdit> レコードについて指定された範囲内を再表示する。
説明　tEHandle で指定された <tEdit> レコードについて，指定された範囲をバックグランドカラーで塗り潰したあとで再表示します。塗り潰す必要がない場合は TMUpDate 3()を使用してください。
<compact> が発生する可能性があります。
●int TMDelete(tEdit **tEHandle);
返り値　0：編集しなかった（V 1. 10より）
1：編集した（V 1. 10より）
負数ならエラーコード。
コール　$A 323
説明　tEHandle で指定された <tEdit> レコードのセレクト部分をカットします。テキストマンスクラップの内容は変わりません。
<compact> が発生する可能性があります。
再表示します。
●int TMEventW(tEdit ** tEHandle, tsevent *EventPtr);
返り値　0：編集しなかった（V1.10より）
1：編集した（V1.10より）
負数ならエラーコード。
コール　ライブラリのみ
機能　イベント発生時の処理( <window> 対応)。
説明　wUpdate の範囲で描画したあとに EventPtr で指定されたイベントレコードにあった処理を行います。teInPort に格納されているのが <window> へのポインタであると仮定して処理しますので，それ以外では使用しないでください。
また wUpdate を操作しますので，WMUpdate() コール中には使用できません。wUpdate が <null region> 以外の場合に，wUpdate と現在の <clip region>が重なった範囲を描画したのちに TMEvent()をコールします。それ以外の場合にはTMEvent()をコールするだけです。
描画した範囲は WMSubRgn() により wUpdate より除かれます。
<compact> が発生する可能性があります。
●int TMNew2(rect *destRectPtr, rect *viewRectPtr, graph *part, tEdit ***tEHdlPtr);
返り値　負数ならエラーコード。
コール　$A 401
機能　新しい <tEdit> レコードを生成し，そのレコードへのハンドルを返す（編集用のメモリを確保する)。
説明　destRect のデスティネーションレクタングルと viewRectのビューレクタングルを指定して新しい<tEdit>レコードを作成します。
<compact> が発生する可能性があります。

## タスクマン(Task Man)

●int TSEndDrag(int mode);
引数　mode　0 以外で元の場所までドラッグする
返り値　負数ならエラーコード。
コール　$A 38 C
機能　ドラッグを終了する。
説明　ドラッグを終了し，ビットイメージのデータを解放します。
mode に 0 以外を指定すると元の場所までドラッグします。
<compact> が発生する可能性があります。
●int TSEventAvail(int eventMask, tsevent *eventrecPtr);
返り値　0：通常のイベント
1：タスク間通信
3：タスク指定のイベント
-8194：ハードウェアエラーによるアボート。
コール　$A 357

機能　タスクマンのイベントで eventMask で指定されたタイプのイベントのうちでもっとも優先順位の高いものを取り出し eventrecPtr に入れて返します。
イベントは取除きません。
<compact> が発生する可能性があります。
注意1　タスクマンのイベントの先読みをします。イベントマスクの指定の仕方はイベントマンと同様です。ほかにタスクがあるときはタスクの切り替えが行われます。
また，イベントレコードにはイベントは格納されていません。
注意2　ハードウェアエラーによるアボートが発生した場合に TSSetAbort() で設定された関数に制御が渡されます。TSSetAbort() が実行されていない場合は exit() で終了してしまいますので，終了処理が必要な場合には，必ずアボート処理ルーチンを登録しておいてください。

●int TSFindOwn(void);
返り値　下位ワード：タスクID。負数なら存在しない。
コール　$A3FE
機能　自分と同じ名前のタスクが存在するかを調べて，そのタスクIDを返します。

●int TSGetDrag(drag ＊＊dragPtr);
引数　dragPtr ドラッグレコードへのポインタを格納する変数。
返り値　負数ならドラッグデータがない。
コール　$A389
説明　ドラッグレコードへのポインタを返します。ポインタはシステム内のドラッグレコードを指していますので，不用意な書き換えは不可です。

●void TSGetEvent(int eventMask, tsevent ＊eventrecPtr);
コール　$A358
機能　タスクマンのイベントを取り除く。
<compact> が発生する可能性があります。

●int TSGetID(void);
コール　$A360
説明　現在動作中のタスクIDを返す。

●int TSGetTdb(task ＊buffPtr, int tskid);
引数　buffPtr タスク管理レコード格納アドレス
tskid　－1で自分のID
返り値　負数ならエラーコード。
コール　$A35B
機能　tskid で指定されたタスクの，タスク管理テーブルを buffPtr に複写します。tskidに負数が指定されたときは，このサービスをコールしたタスクのタスク管理テーブルが複写されます。変更して設定する場合は TSSetTdb() をコールします。

●int TSGetWindowPos(void);
返り値　ウィンドウの左上の座標。
コール　$A35E
説明　次にウィンドウをオープンする場所をランダムに発生します。

●int TSSetAbort(void (＊process)(), long param);
返り値　前の処理ルーチンへのポインタが返ります。
コール　ライブラリのみ
機能　ハードウェアエラーによるアボートが発生した場合に process で指定した関数に制御が渡されます。
(＊process)(-8194, param)の形式で呼び出されます。
アボート処理ルーチンが設定されてないと exit() で終了してしまいますので，終了処理が必要な場合には，必ずアボート処理ルーチンを登録しておいてください。

●int TSTakeParam(LASCII ＊command, rect ＊wrectptr, char ＊namePtr, int mode, char ＊destbuff, char ＊＊ptr);
引数　command　コマンドライン
wrectptr　ウィンドウ位置指定格納用（0で指定しない）
namePtr 文字列格納用(0で指定しない)
mode　0：Ptr を確保しない
2：destbuff あり
destbuff アドレステーブル格納バッファ
unsigned char ＊＊ptr　確保したポインタ
返り値　パラメータの有無。
Bit0：ONでウィンドウの位置の指定あり
Bit1：ONで文字列あり
負数ならエラーコード
コール　$A3EA
説明　command で指定されたコマンドラインをスペースの区切りで分け，そのアドレステーブルを作成して返します。
wrectptr が指定されている場合に，ウィンドウ位置指定のオプションを取り出し座標に変換して格納します（8バイト必要）。
namePtr が指定されている場合に，最後に指定された文字列を格納します。
mode によって処理が異なります。
・mode = 0　アドレステーブルを返しません。パラメータ抽出後廃棄されます。
・mode = 1　作成したアドレステーブルをポインタにコピーして，そのポインタを返します。<compact> が発生する可能性があります。
・mode = 2　作成したアドレステーブルを destbuff で指定されたアドレスにコピーします。

## ウィンドウマン(Window Man)

●int WMDispose(window ＊Window);
返り値　終了コードが返ります。終了コードが負の場合はエラーです。
コール　$A1FC
機能　指定のウィンドウを消去します。
説明　WMOpen(WMRefer)で確保したメモリを解放し指定の window をウィンドウリストから削除し，もし画面に表示されているなら画面より消去します。
WMOpen(WMRefer)で Window に0を指定したときに使用してください。
<compact> が発生します。

●int WMDrawGBox(window ＊Window);
返り値　終了コードが返ります。終了コードが負の場合はエラーです。
コール　$A20C
説明　グローボックスを書きます。
<compact> が発生します。

●int WMFind(point_t pt, window ＊＊win);
引数　win には window への pointer のアドレスを入れてください。座標の当たっているウィンドウへの pointer が入ります。
返り値　パードコードが返ります。負の場合はエラーです。
コール　$A1FD
機能　指定の座標がどのウィンドウのどの部分にあるか調べます。
説明　指定の座標（グローバル座標系）が存在する window への pointer を返します。またウィンドウのどの部分に当たるかをウィンドウパートコードで返します。

●window ＊WMOpen(window ＊Window, rect ＊bounds, LASCII ＊title, int visble, int wDefID, window ＊behind, int cBox, long taskID);
返り値　window pointer 0のときはエラーが発生しています。errno にエラーコードを参照してください。
コール　$A1F9
説明　Window が0のときメモリマンを使用してメモリを確保します。
一次的なウィンドウの場合スタックに window を確保して(link 命令など)から呼び出すと heap が細切れになりません。
bounds はウィンドウの画面上の位置と大きさを指定します(グローバル座標系)。指定のレクトアングルは wInside になります。(正確には，ウィンドウ定義関数によって決まる)
title はウィンドウのタイトルになります。ウィンドウ定義関数によっては無視されます。
visble は描画許可フラグです。0以外でウィンドウは表示される可視状態になります。
wDefID はウィンドウ定義関数の resource ID を16倍したものです。下位4ビットは wOption です。標準ウィンドウは32です。たとえば，32＜＜4＝512となります。
behind はウィンドウの順番を指定します。
この作成されたウィンドウは behind の後ろに配置されます。ただしこの値が0のときは一番後ろに，－1のときは一番前にウィンドウを開きます。
cBox はクローズボックスの有無を示すフラグです。0以外のときクローズボックスを表示します。ただしウィンドウ定義関数によっては無視するものもあります。
taskID はSXシェル上で動作するアプリケーションなら自分のタスク番号を入れてください。SXシェル上で動作しないものなら自由にお使いください。
通常不可視属性でウィンドウを開きスクロールバーなどを配置して，その後ウィンドウを可視属性にします。こうすることで初めからスクロールバーが表示されたウィンドウが出現したように見えます。
<compact> が発生します。

| resource ID | |
|---|---|
| 32 | 標準ウィンドウ |
| 36 | プレーンウィンドウ |
| 48 | 標準ウィンドウ（グラフィックサポート） |
| 50 | 標準ウィンドウ（グラフィックサポート，クローズボックスのみ） |

●int WMSelect(window ＊Window);
返り値　終了コードが返ります。終了コードが負の場合はエラーです。
コール　$A1FE
機能　ウィンドウをアクティブにします。
<compact> が発生します。

●int WMShow(window ＊Window);
返り値　終了コードが返ります。終了コードが負の場合はエラーです。
コール　$A207
機能　ウィンドウを可視属性にします。
説明　不可視属性のウィンドウを可視属性にします。
ウィンドウを書き直します。
<compact> が発生します。

●int WMSize(window ＊Window, point_t pt, int uFlag);
返り値　終了コードが返ります。終了コードが負の場合はエラーです。
コール　$A203
機能　ウィンドウの大きさを設定します。
説明　size は x，y にそれぞれウィンドウの幅，高さを設定してください。
uFlag が偽（0）のとき，指定のウィンドウだけは wUpdate を更新しません。
<compact> が発生します。

●int WMUpdate(window ＊Window);
返り値　終了コードが返ります。終了コードが負の場合はエラーです。
コール　$A20D
機能　ウィンドウの書き直しを開始できるようにします（アップデート状態）。
説明　アップデートイベント発生時に使用してください。具体的には，ウィンドウは Update でクリップされそのリージョンにしか書き込めなくなります。したがってこの呼び出しを行ったあと次の WMUpdtOver() を実行するまでの間，Update を変更するような操作を行ってはいけません。
<compact> が発生します。

●int WMUpdtOver(window ＊Window);
返り値　終了コードが返ります。終了コードが負の場合はエラーです。
コール　$A20E
機能　指定のウィンドウのアップデート状態を通常の状態に戻します。
<compact> が発生します。

プログラミングスタイルは変わるのか？

# ウィンドウとオブジェクト指向

SX-WINDOW

Izumi Daisuke 泉 大介

ウィンドウプログラミングとなると，しゃしゃり出てくるのが「オブジェクト指向」という言葉です。Smalltalkの基本概念を中心に，どのようなメリットがあるのか，本当に使わなければならないのかを考えてみましょう。

オブジェクト指向，あるいはOOPS(Object Oriented Programing System)という言葉が一時流行になりました。グラフィックシステムやウィンドウシステムと相性がいいといわれる考え方です。そのためか，MS-WINDOWS3.0の登場に時期をあわせるかのように再び目につくようになりました。C++，G++といった，オブジェクト指向のプログラミング言語のアナウンスも盛んです。オブジェクト指向とはどのようなシステムなのか，本当に使いやすいものなのか，それをここで考えてみたいと思います。

## データ＋関数＝オブジェクト？

オブジェクトとはいったいなんなのか。これはオブジェクト指向に対する究極の質問です。実際，なにをオブジェクトとするかを決めるだけで，オブジェクト指向の言語処理系などいくつでも作れるのです。いささか乱暴ですが，ここではウィンドウシステムにオブジェクト指向を導入する際によく使用される例を引用して，SX-WINDOWの画面上に表示されるウィンドウやボタン，ダイアログなどをオブジェクトとすることにしましょう。

**図1　拡張された構造体**

| ウィンドウレコード |
|---|
| アウトサイドリージョン |
| インサイドリージョン |
| タイトル |
| などなど |
| ウィンドウ操作関数 |
| new |
| open |
| close |
| dispose |
| move |
| size |
| などなど |

ウィンドウを例とすれば，このオブジェクトは画面のどこに表示され，どの程度の大きさを持つものなのか，タイトルはなんなのか，スクロールバーはあるのか，といったデータを保持しています。これはC言語やアセンブリ言語で構造体を使ってウィンドウを定義する場合と同じです。

これらの情報に加えてこのオブジェクトは，表示する座標やウィンドウのサイズを設定する方法，画面に自分をオープンするための方法，アップデートする方法，クローズする方法をも保持しています。つまり，データとそのデータを扱う関数をすべて内蔵しているというわけです(図1)。

さてここで，ウィンドウの表示座標やウィンドウサイズなどのデータを，直接参照したり変更したりすることができるのは，オブジェクト内に一緒にパックされているウィンドウ操作関数だけだ，と規定するとします。ほかにこれらのデータを変更する手段はないのですから，プログラムは用意された関数を使ってデータを変更するしかありません。ウィンドウのデータは，プログラムのほかの部分から完全に隠されてしまうことになります。

このような機構を導入するメリットは，次の例でおわかりいただけると思います。仮にマウスの左ボタンが押されたら小さく，右ボタンが押されたら大きくなるウィンドウを考えてみましょう。ウィンドウオブジェクトを導入しない普通のプログラミングでは，「マウスの左ボタンが押された」というイベントを処理するルーチンで，ウィンドウを小さくするように構造体のデータを変更することでしょう。

「マウスの右ボタンが押された」というイベントを処理するルーチンでは逆に，ウィンドウを大きくするように構造体のデータをセットし直すはずです。もしもウィンドウ構造体に変更があった場合には，プログラム中のこれらの部分を探し出して，片っ端から変更しなければなりません。

ウィンドウオブジェクトを利用したプログラミングなら，これらのルーチンではサイズ変更の関数呼び出ししか行われていませんから，新しいオブジェクトに差し替えるだけでこと足ります。関数の呼び出し方さえ統一しておけば，データ構造がどのように変更されようと外部にはまったく影響を与えないで済むのです。

次に，このように内部にパックされてしまった関数を実行する方法について触れておきましょう。オブジェクト指向言語の最右翼であるSmalltalkでは，メッセージパッシングと呼ばれる方法がとられています。ウィンドウオブジェクトをwindowという名前で作成したとすると，これを画面に表示するには，

```
window open
```

のようにオブジェクトに「開け」というメッセージを送ればいい，というのです。

メッセージを受け取ったオブジェクトは自分が抱え込んでいる関数を調べ，対応する関数を使って実際にウィンドウを開くことになります。つまりオブジェクトは，データとそれを処理する関数の単なる集合体ではなく，ユーザーが発するメッセージに応えることのできるインテリジェントな物体ととらえているのです。

SX-WINDOWの画面に表示されるものにはダイアログやコントロールなどがありますが，ユーザーはどうすればダイアログを開くことができるのか，どうすればコントロールを開くことができるのかといった情報を知っている必要はありません。dialogというオブジェクトを開くには，

```
dialog open
```

とすればOKですし，controlというオブジェクトを開くには，

```
control open
```

とすればいいのです。実際にどのような手

順を踏まなければならないのかはオブジェクトが知っているというわけです。SX-WINDOWでいうなら，このopenというメッセージに応えるために，windowならWMOpen, dialogならDMOpen, controlならCMOpenというシステムコールが利用されることでしょう。しかし，ユーザーがこのことを関知している必要はありません。単純にopen，これでいいのです。

グラフィックシステムやウィンドウシステムには，このように同じ名前でくくることのできる操作というものが少なからず存在します。オブジェクトをアクティブにする，描画する，消すなどといった同じような操作をひとつのメッセージを覚えるだけで利用できるということは，かなりメリットが大きいといえるでしょう。

## 性質の継承と差分プログラミング

データの陰蔽とともにオブジェクト指向のメリットとされる特徴として，オブジェクト間には親子関係があり，親の持っている性質は子に継承されるという点を挙げることができます。このため子のオブジェクトを作成するときには，親と異なる差分の部分だけを用意すればこと足ります。

例として画面上の点を表すpointというオブジェクトを考えてみます。pointはその性格上XY座標をデータとして持っています。そして，ここに座標をセットするsetx:, sety:という2つのメッセージに応えることができるとしましょう。これらのメッセージは,

```
point setx:10
point sety:20
```

のように使うことにしておきます。

画面上の線を表すlineというオブジェクトは，始点と終点の2つの座標を必要とします。このlineオブジェクトをpointを親として作成してみたのが図2です。lineオブジェクト用追加部分には始点を保持するデータも始点をセットするメッセージも用意されていません。親のものを利用できるからです。lineオブジェクトに,

```
line setxe:100
```

とメッセージを送れば追加部分のメッセージが利用されて終点のX座標100がセットされるのはご想像のとおりです。

```
line setx:10
```

とした場合には，setx:というメッセージが追加部分にはありませんから，lineオブジェクトは自分の親のメッセージを利用します。そしてpointオブジェクトのsetx:メッセージが使われてpointオブジェクト部分のX座標にデータがセットされることになります。こうして，終点座標用データとそれにデータをセットするためのメッセージを用意するだけで，始点・終点のデータを持ち，それぞれにデータをセット可能な新しいlineオブジェクトが作成できるわけです。

このように，差分を記述するだけで親の持っている情報をそっくり受け継いだ新しいオブジェクトを作成できることは，プログラミングの手間を大幅に省くことになります。

もちろん，このままでは画面にpointやlineを表示することはできません。drawなどといったメッセージが必要となることでしょう。pointへのdrawメッセージとlineへのdrawメッセージは当然異なります。オブジェクトは自分の中にメッセージを見つけたら親のメッセージを利用することはしませんので，双方にdrawメッセージが用意されていても混乱はありません。

さらにlineオブジェクトの子としてrectangleオブジェクトを作成する場合には，lineが始点と終点の座標を扱うことができますので，drawメッセージを作成するだけで表示できるようになるわけです。

このようにしてオブジェクトの階層構造ができあがるわけですが，オブジェクト指向言語では，それぞれの階層を「クラス」という名で呼んでいます。pointクラス，lineクラスというわけです。そして，親のクラスはスーパークラス，子のクラスはサブクラスと呼ばれます。

クラスはC言語やアセンブリ言語でいうところの「ソースリスト」で，どのようなデータを保持することができるのか，どのようなメッセージに応答することができるのかといった設計図です。Smalltalkなどでは，クラスはエディタを使って作成し，然る後にこのクラスを「コンパイル」なり「アセンブル」なりして，実際にデータをセットしたり，画面に表示させたりする実行ファイルに相当するもの（インスタンスと呼ばれる）を作成するようになっています。

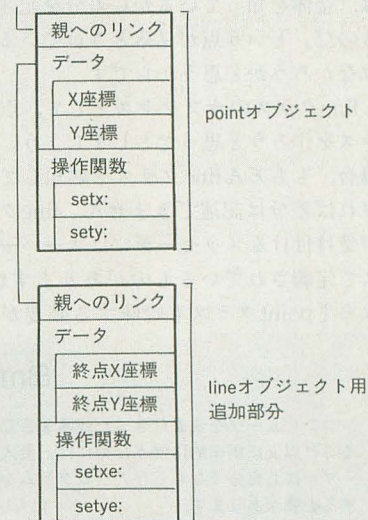
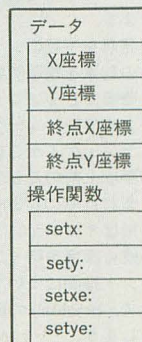
また，オブジェクト内に包含された関数はメソッドと呼ばれています。このあたりの用語はオブジェクト指向関係の書籍を読むときには必ずといっていいほど登場する言葉ですので，覚えておかれるといいでしょう。

ソースリスト→実行ファイル，つまり，クラス→インスタンスの区別がはっきり分かれているのは言語仕様として美しくないとばかりに，両者の性格をあわせてしまった言語も存在します。こういった言語ではメソッドを登録するためのメッセージが用意されており，試行錯誤しながらプログラミングを進めていけるようになっています。ひとつのオブジェクトがクラスであり同時にインスタンスでもあるわけですから，データをセットすることもできれば，そのオブジェクトを親とする新しいオブジェクトを作成することも可能です。

また，関数名をオブジェクトだとして，関数型言語にオブジェクト指向を導入する例もあります。オブジェクト指向言語の枝葉末節に入っていったのではきりがありません。このあたりでやめておくことにします。

## オブジェクト指向は使えるのか

少々低級なプログラミングをやった経験のある方なら，「オブジェクトにメッセージ

**図2 lineオブジェクトを表す構造体**

2) 次のようなオブジェクトを
作成したかのように扱える

を送る」「メッセージを探す」「なければ親のメッセージ使をう」といった機構が抱えるオーバーヘッドの大きさにすでに気づいておられると思います。

2＋3

が「2」というオブジェクトに「＋」というメッセージを送る。パラメータは3。などと実現されているかと思うと寒気がするという意見には耳を傾けるべきものがあります。

一切がっさいをすべてメッセージを送ることで処理しようというのは，「美しくはあるが，あまり実用的とはいえない」でしょう。こういった試みは，Smalltalk以降あまり見られなくなってきています。

代わって，オブジェクト指向のいいところだけを取り込もうという姿勢が現在の大勢です。データの陰蔽化はプログラムのモジュール化をいっそう促進させます。保守が容易になることうけあいです。差分プログラミングはプログラム作りの労力を軽減することでしょう。そう思われています。

ところが個人的には後者にはいささかの疑問符をつけざるを得ません。というのは，「全体を知っているから差分を記述できるのだ」という点が見落とされているのではなかろうかと思うからです。

lineクラスのサブクラスとして，点線クラスを作ろうと思ったとしましょう。この場合，もちろんlineクラスを熟知していなければ差分は記述できません。lineクラスが受け付けるメッセージにはスーパークラスで定義されているものがありますから，きっとpointクラスも吟味する必要があるでしょう。

pointクラスもなにかのサブクラスですから，これも遡る必要があります。こうしてクラスの階層構造のかなり上まで知っていて初めて，記述すべき「差分」がどのようなものなのかがわかるのです。

また，自分が作成しようとしているクラスを，既存のどのクラスのサブクラスとすれば効率よく差分を記述できるのかという点も同様です。この場合はシステム全体を知らないと，無駄なプログラムを強いられることになるでしょう。システムが大きくなればなるほど，初めてプログラミングに取り掛かろうとする者への負担は大きくなります。

## そしてHyperCardは

HyperCardはMacintoshに用意されているプログラマブルカード型データベースです。ユーザーはボタンなどをカード上に自由に配置して，「押されたら～する」とプログラムしておくことができます。より正確にいうなら，HyperCardはユーザーが自由にカード型データベースを作成できるようなプラットフォームを提供するものといえます。

カードに絵を描いておいて，適当な場所にボタンをちりばめておく。ボタンには，「押されたら別のカードを表示する」とプログラムをつけておけば，簡単なアドベンチャーゲームが作成できます。「それは取れません」と音声を発するように指示しておくのも面白いでしょう。

このHyperCardがオブジェクト指向を取り入れているのです。カード上に配置されたボタンには，前述のようなプログラムを書いておくことができます。つまり，ボタンは押されたらなにをすればいいのかを知っているわけです。ボタンが押されると，このボタンに「押された」というメッセージが送られるようになっています。

カードは平面に表示されていますが，実際はボタン，カード，バックグラウンド(背景)，スタック(カードの集合体)，ホームスタック，HyperCardの順に階層構造をしています。そして，より上位の階層で定義された動作は，自動的に下位の階層に引き継がれるようになっています。すべてのカードに共通する動作を定義したければ，バックグラウンドにその動作を書き込んでおけばいいわけです。

発生するメッセージは種々ありますが，ボタンなりカードなりが反応したいメッセージにだけ応答するようにしておけば，それ以外のメッセージは自動的により上位の階層で処理してくれます。つまり，階層による動作の継承と差分の記述が可能になっているわけです。

それぞれのメッセージに対する動作は英語の平文のような言語で記述するようになっており，オブジェクト指向を思わせるものはありません。HyperCardは限られた環境にオブジェクト指向を取り入れ，実際の視覚効果とオーバーラップさせるかのように実にうまく実装している例といえるでしょう。

＊　＊　＊

オブジェクト指向の概略と，そのメリット・デメリットについて考えてきました。ウィンドウシステムの画面上には，ウィンドウ，ダイアログ，ボタン，アイコンといった，限られた要素しか存在していません。また，SX-WINDOWが処理するイベントの数を見てもわかるように，発生するイベントの数も限られたものです。画面上の限られた要素(クラス)と発生する限られたイベント(メッセージ)。こういった環境にうまくオブジェクト指向を持ち込めれば，性質の継承と差分プログラミングを中心とした効率的な環境を構築できることはHyper Cardの例からも明らかです。

現在のSX-WINDOWでは，簡単なバッチ処理程度のこともできません。うまくやれば，マウスで簡単に指示できるオブジェクト指向のプログラムレスプログラミング環境といったものが実現できるのではないか。そんな夢を抱いています。

### Smalltalkの環境

コンピュータを決まりきった仕事を自動化すること以上に創造的に使うためには，素人のユーザーにも自分でなんらかのプログラムを作成する必要があります。

しかし，いま流行のC言語は本来プログラマがミスを犯すことなど想定されていない言語でしたし，お馴染みのBASICは初心者にも専門家にもプログラムの作成が不可能な言語とさえいわれることさえもあります。

「ユーザーが自分の発想を手軽にシミュレートできる」そして「その結果を視聴覚に訴えるかたちでわかりやすく確認できるプログラミング環境」こそがパーソナルコンピュータのあるべき姿ではないか，アラン・ケイがこのように考えていたのはもう20年以上前のことです。

こうした考えから生まれたものがSmalltalkという言語です。その特徴はオブジェクト指向であったことと，独自の統合環境を持っていたことでしょう。オブジェクト指向については本文で触れられていますので，Smalltalkの環境についてざっと見ていきましょう。

ビットマップディスプレイ，キーボードとマウス（3ボタン）を駆使したシステムと画面構成は，マルチウィンドウもあいまって現在でも古さを感じさせません。

さまざまなフォントセットを持っており，それらがプログラムの表記に使用されます。見た目にわかりやすいプログラムを追求した結果でしょう。しかし，エディタでマルチフォントを使い分けたプログラムを書くというような必要はありません。Smalltalkは言語本体のほかにシステムブラウザ，デバッガ，スペルチェッカやグラフィックエディタまで内包したシステムで構成されています。ユーザーはマルチウィンドウで表示されるテンプレート（プログラムの枠組み）をキーボードとマウスで埋めていくというものです。

バイトコードインタプリタを介したり，徹底したオブジェクト指向でのオーバーヘッドがあったりと言語としてのSmalltalkにはつらいものがあります。しかし，環境としてのSmalltalkはいまなお輝きを失っていません。

# 第115部　パズルゲームLINER

●パズルゲームLINER

久しぶりにじっくりと腰を落ち着けて取り組めるパズルゲームの登場です。ルールは極めて簡単。用意された9×9のマス目の中で，一筆書きをすればいいのです。とはいってもそこはそれ，パズルゲームですから，ルールがあります。マスの中には数の書き込まれたものがあり，01が表示されているマスからスタートしたプレイヤーは，必ず第ｎ歩目にｎと表示されたマスを通らなければなりません。

数の書き込まれたマスの数が多いうちは，「あと5歩でこのマスに入ればいいのだから……」と比較的簡単に歩を進めていくことができるのですが，ヒントが少なくなるにつれ，「あと20歩でこのマスに……」と難しさを増していきます。どうすれば歩数を稼げるのか，これを考えるのがこのゲームのポイントです。単純なルールながら結構考えさせられるゲームです。

プログラムは面クリアシステムではなく，自分の好きな面に挑戦できるようになっています。また，新しい面の作成方法も解説されていますので，友達どうしで問題を出し合うといった楽しみ方もできるようになっています。自分で面を作成してみると，いかにヒントを出すことによって，難しくできるかの吟味が結構やっかいです。安易にやると複数のルートができてしまいます。ゲームプレイだけでなく面作成でも悩ませるゲームです。

●開発言語の歴史

S-OS “MACE” に始まり現在までさまざまな開発ツールが提供されてきました。そして，今回のように独自に開発したツールを含めるとかなりの数になることでしょう。

試しに現在までに発表された開発ツールをまとめてみます。まずはアセンブラです。エディタアセンブラ「ZEDA」に始まって高速化，分割アセンブル可能となった「ZEDA II(改良版)」に「ZEDA-3」，そして「改造版ZEDA」の作者である瀧山氏による「REDA」，大貫氏制作の「これでもアセンブラか？」と思えるほどの超多機能アセンブラ「OHM-Z80」，石上氏がSmall-Cを移植するために開発したリロケータブルアセンブラ「WZD」の6種類があります。

そして，開発言語についても既製のものからオリジナルまで，かなり幅広く提供されています。以下，列挙してみると「Lisp-85インタプリタ」，「Prolog-85」，「magiFORTH」，「FuzzyBASIC」，「FuzzyBASICコンパイラ(石上版，奥村版)」，「SLANG」，「TTI」，「TTC」，「STACK」，「REAL」，「Small-C」があります。また，いままでの視点とは違った新しい発想の言語もこれからどんどん発表されると思います。活用するのは読者の皆さん，あなたたちです。せっかく提供された環境ですから，ぜひがんばって使いこなしていきましょう。

●S-OSの系譜(28)

S-OSでマシン語プログラミングを行ううえで必要不可欠なアイテム，それがアセンブラとデバッガです。これはS-OS第2号である1985年の7月号で掲載されました。S-OSのシステムが掲載された第1号である1985年の6月号とともに，いち早くバックナンバー切れを起こした号です。

このため，「S-OSに参加したいのだがアセンブラ・デバッガがない」という，新規参入される方にはつらい状況が続いていました。幸いアセンブラのほうは瀧山氏の手により，アセンブルスピードを大幅に向上させたZEDA-3が1987年6月号で発表され，状況は若干改善されたのですが，デバッガのほうはあいかわらずサポートも再掲載もされないままでした。

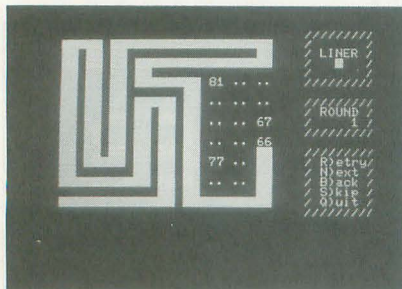
1988年4月号はユーザー待望の新しいデバッガTRADEが登場した号です。メモリをダンプしたり，レジスタの内容を表示する，任意の場所でマシン語プログラムの実行を中止できるようにするといったZAIDが備えている機能に加え，Z80の命令をひとつずつレジスタの内容を表示しながら実行していくというZ80TRACER(1986年6月号掲載)の機能を包含しています。さらに，プログラムのアセンブルを行った直後など，ラベルテーブル(特殊ワークエリア)が破壊されていない状況では，設定したラベルを逆アセンブル時に表示するシンボリックデバッグ機能も備えています。ラベルの値は自分で設定できるようになっており，特殊ワークが破壊された後でのシンボリックデバッグにも配慮されていました。

サイズは8Kバイトとデバッガとしては少々巨大になっています。このため，オフセットをつけて生成したプログラムをデバッグするための機能まで搭載されていました。この状態ではブレイクポイントを設定してのデバッグはできませんが，嬉しい配慮です。

全機種共通
S-OS"SWORD"要

# LINER

Satou Yoshihiro
佐藤 義弘

**今月はちょっと手軽なパズルゲームの登場です。ルールは画面全体をラインを引きながら埋め尽くすだけ。さあ，こたつにミカンを用意して遊びましょう。**

このゲームは，なんとなく考えていたらなんとなくできてしまったという，本当にお手軽なパズルゲームです。開発も順調だったので作っている本人も楽しんでプログラミングすることができました。強いていえばいちばん悩んだのが，面構成を考える段階です。適当に作ったのはいいけどなぜか解けない。はっきりいってデバッグするよりも悩んだといっていいでしょう。

ということで，こんな調子で作ったプログラムですから軽い気持ちで入力し，楽しんでくれれば作者としてはとてもうれしいです。リストも手頃な大きさですし，ちょっとした暇があればすぐにでも入力できるでしょう。

## 遊び方

ルールは簡単で画面上にある数字の01から81までひと筆書きの要領で線を引いていくだけです。01のある場所からスタートして，ひと駒移動するごとに数が増えていき，逆にひとつ戻ると数が減ります。

そして，画面上にある数字と自分の移動数が同じ場合にのみ，その数字を通過することができます。こういった要領でうまく画面上にある数を合わせながら線を引いていき，81にたどりつくとラウンドクリアとなります。ルールは簡単でもなかなかクリアはできないでしょう。特に通過する数字が少ない場合には，それだけルートの選択が多くなり，途中までうまく進んでも曲がり方ひとつ間違うと，ほとんど最初からやり直しになってしまいますから注意しましょう。

全部で15面，クリア不能な面はありませんからがんばってください。

## 入力方法

入力方法はいつものとおり，MACINTO-Cなどのマシン語入力ツールを使ってリスト1のダンプリストを打ち込んでください。打ち込み終わったらチェックサムが合っているかどうか確認したあとに，

#S LINER：5000：5932：5000

のようにしてセーブしてください。

実行は，

#J5000

です。起動後は，すぺぺぺとタイトルが描かれキー入力待ちになります。ゲームを始めたいときにはスペースキーを押してください。ここでしばらくキー入力を押さずにいると1面のオートデモが始まります。先ほどの説明でいまいちルールがのみ込めなかった人は，このデモを見てくれれば理解できると思います。一応，ラウンド1の模範解答を見せてくれます。

## 操作方法

操作方法は，カーソルキー，2,4,6,8キー，そしてI,J,K,Lキーで上下左右に線を引いていくことができます。そのほか，ゲーム中では以下のキーを使用することができます。

R……現在プレイしている面をやり直します
N……次の面に進みます
B……ひとつ前の面に戻ります
S……指定した面に移ります。移動キーでプレイしたい面を決め，リターンキーで決定します
Q……ゲームを終了してS-OSモニタに戻ります

ゲームにはこれといってペナルティが存在しません。適当にあっちこっちの面をつっつきながら遊んでください。

## プログラムについて

これといって特殊なことはしていませんが，ソースリストを見ていただければわかるとおり，いわゆる普通のアセンブラを使わずにオリジナルのアセンブラ「ZASM」を使用しました(十分特殊だって？)。ですからソースリストは本当に参考程度のものとなっています。

ちなみに，このアセンブラは大貫氏が制作した「OHM-Z80」には及びませんが，それなりにマクロが使えるためなかなか重宝しています。特にループでのローカルラベルが使用できるので，ラベルの使用を少なくできソースリストをきれいにまとめることができるのです。

また，このアセンブラの発表の予定はいまのところありません。現在のところアセンブラだけでも6種類ありますしね。

話変わって，オリジナル面の作成についてちょっと話しておきましょう。ソースリストを見ればわかるでしょうが，417行にあるRDATA (5474H)というラベル以降に15面分のデータが格納されています。1面あたり81バイト，内容もシンプルなもので0が空白，それ以外の値は数字が書かれます。注意してほしいのは，スタート地点である01，そしてゴール地点の81を忘れずにデータセットしてください。

15面すべて遊んでしまったら，ぜひ自分でオリジナルデータを作って友達でもはめて遊んでしまいましょう。

＊　＊　＊

最近システム関係のプログラムが多いですね。それはそれで開発環境が充実していいことだと思いますが，こういった息抜き的に遊べるパズルゲームもほしいと思うんですけど，どんなもんでしょう？　アセンブラにかぎらず，現在までにさまざまな言語がありますし，そういったものを使えば比較的簡単に作れます。しかもパズルゲームならそれほど重くならず，手軽にプログラム作成ができるはずですから。

では，S-OSユーザーズクラブの森さん，および会員の皆さん，そしてS-OSを使用しているユーザーの皆さん，これからもS-OSをどんどん盛り上げていきましょう。

リスト1

```
5000 C3 14 50 0F 2E 2E 00 7B : 0D
5008 7B 00 7B 7B 00 7B 00 20 : 0C
5010 20 00 20 00 31 00 40 CD : 7E
5018 02 53 CD 7D 52 3E AF 32 : 10
5020 06 42 3E 01 32 07 42 CD : CF
5028 64 52 31 00 40 CD 0E 52 : 54
5030 CD 64 50 3A 05 42 B7 28 : E1
5038 F7 CD 2D 51 11 07 50 CD : 77
5040 E5 1F 11 25 54 CD BA 51 : 66
5048 3A 06 42 B7 28 05 CD C0 : F3
5050 52 18 C7 CD D0 1F FE 1B : 06
5058 CA 86 51 FE 20 20 F4 CD : A0
5060 44 51 18 B6 CD 2D 51 2A : D8
5068 00 42 24 36 FF 3A 04 42 : 1B
5070 CD DD 51 21 22 03 CD 1E : 2C
5078 20 CD 94 51 CD 8F 50 30 : AE
----------------------------------
SUM: FA 2C 30 98 60 0E 31 61 363A

5080 1D 4F 7E 23 B7 C8 5E 23 : 0D
5088 56 23 B9 20 F5 EB E9 21 : 3C
5090 29 53 01 0C 00 ED B1 37 : 5E
5098 C0 79 E6 03 4F C9 ED 43 : 6A
50A0 02 42 21 48 53 09 09 4E : 60
50A8 23 46 2A 00 42 09 11 04 : F3
50B0 42 7E B7 28 0E E6 7F 4F : 61
50B8 1A 3C B9 28 0C D6 02 B9 : D4
50C0 28 29 C9 1A C6 81 77 D6 : C8
50C8 80 12 E5 CD 2D 51 E1 22 : C5
50D0 00 42 11 07 50 CD E5 1F : 7B
50D8 11 0A 50 21 0D 50 CD 12 : C8
50E0 51 3A 04 42 FE 51 C0 32 : 12
50E8 05 42 C9 24 7E 25 B7 C8 : 56
50F0 1A 3D 12 E5 CD 2D 51 2A : C3
50F8 00 42 24 36 00 25 7E CB : 0A
----------------------------------
SUM: 06 02 EB 7A 43 EE D0 30 E233

5100 7F 28 02 AF 77 CD D4 51 : C1
5108 E1 22 00 42 11 0F 50 21 : D6
5110 12 50 D5 E5 21 50 53 ED : CD
5118 4B 02 42 09 09 09 09 EB : 9E
5120 CD E5 1F E1 D1 CB 41 20 : AF
5128 01 EB C3 E5 1F 3A 00 42 : 2F
5130 26 FD 24 24 D6 0A 30 FA : 75
5138 C6 0A 6F 87 85 6F C3 1E : 9B
5140 20 3E FF 21 3E 01 CD 78 : 02
5148 51 D8 C3 2A 50 3A 07 42 : E9
5150 32 08 42 CD E2 51 CD 9A : E3
5158 51 CD 8F 50 38 08 E6 02 : 25
5160 3D CD 78 51 18 ED FE 0D : E3
5168 CA 2A 50 FE 1B 20 E4 3A : 9B
5170 08 42 32 07 42 C3 E2 51 : BB
5178 21 07 42 86 37 C8 4F 3A : 78
----------------------------------
SUM: 9B 9E 5D 94 51 DF 4E EC AB3D

5180 03 50 B9 D8 71 C9 CD E2 : CD
5188 1F 0C 51 75 69 74 0D 0D : E8
5190 00 C3 FA 1F 3A 06 42 B7 : 15
5198 20 0C CD 21 20 FE 61 D8 : 71
51A0 FE 7B D0 D6 20 C9 21 00 : 29
51A8 80 2B 7C B5 20 FB CD F4 : B8
51B0 52 2A 09 42 7E 23 22 09 : 93
51B8 42 C9 EB 5E 23 56 23 EB : DB
51C0 FE 24 CD 1E 20 1A 13 B7 : 11
51C8 C8 FE 02 38 ED 28 F2 CD : D4
51D0 F4 1F 18 F1 B7 20 06 11 : 0A
51D8 04 50 C3 E5 1F 57 0E 30 : B0
51E0 18 11 21 22 09 CD 1E 20 : 80
51E8 3A 07 42 57 0E 20 1E 64 : 8A
51F0 CD FB 51 1E 0A CD FB 51 : 5A
51F8 42 18 0C 06 00 7A 04 93 : 7D
----------------------------------
SUM: 73 80 7B 81 19 6B 04 93 E8B3

5200 30 FC 83 57 05 28 02 0E : 43
5208 30 78 81 C3 F4 1F CD E2 : AE
5210 51 AF 32 05 42 3C 32 04 : EB
5218 42 21 00 01 CD 1E 20 2A : 99
5220 07 42 26 00 2B 54 5D 29 : 74
5228 29 19 29 29 29 29 19 11 : 10
5230 74 54 19 EB 21 0A 40 0E : 45
5238 09 06 09 24 36 00 25 1A : B1
5240 77 FE 01 20 03 22 00 42 : FD
5248 D9 CD D4 51 D9 CD F1 1F : 81
5250 13 23 10 E7 CD EE 1F 23 : 2A
5258 06 1B CD DF 1F CD EE 1F : C6
5260 0D 20 D6 C9 3E 0C CD F4 : D7
5268 1F 11 77 53 CD BA 51 21 : F3
5270 00 40 36 FF 11 01 40 01 : C8
5278 FF 00 ED B0 C9 3E 0C CD : 7C
----------------------------------
SUM: 34 73 C9 5A 60 D7 64 06 FF40

5280 F4 1F CD D0 1F B7 20 FA : A0
5288 21 0B 42 22 09 42 11 2E : 1A
5290 54 CD BA 51 11 3F 54 CD : 9D
5298 E5 52 38 24 D5 CD F4 52 : 7B
52A0 C5 CD 1E 20 CD E2 1F 6F : 0D
52A8 1D 00 11 00 20 1B 7A B3 : 96
52B0 20 FB 3E 7B CD F4 1F CD : 81
52B8 CC 52 C1 10 E0 D1 18 D7 : 8F
52C0 01 00 30 CD F4 52 0B 78 : C7
52C8 B1 20 F8 C9 3E 2D CD D3 : 9D
52D0 52 3E 25 CD D7 52 3D CB : B3
52D8 11 D0 32 DD 52 00 C9 EB : F6
52E0 5E 23 56 23 EB 1A 13 3C : 4E
52E8 37 C8 3D 28 F2 4F E6 0F : 9A
52F0 47 A9 4F C9 CD D0 1F FE : C2
52F8 1B CA 86 51 FE 20 CA 1E : C2
----------------------------------
SUM: 28 EF 16 B7 AB F1 09 75 F028

5300 50 C9 3E 0C CD F4 1F 3E : 81
5308 28 CD 30 20 11 60 53 21 : 2A
5310 0B 42 1A 13 B7 C8 47 E6 : 26
5318 0F 4F A8 0F 0F 0F 0F 47 : 89
5320 79 C6 30 77 23 10 FC 18 : 2D
5328 E9 32 36 38 34 4B 4C 49 : 9D
5330 4A 1F 1C 1E 1D 52 2A 50 : 8C
5338 4E 44 51 42 41 51 53 4D : 57
5340 51 51 86 51 1B 86 51 00 : 6B
5348 FF FF F6 FF 01 00 0A 00 : FE
5350 1D 1D 1D 00 1E 1D 1D 00 : AF
5358 00 00 00 00 1F 1D 1D 00 : 59
5360 46 58 14 42 34 78 16 62 : 18
5368 16 68 66 12 54 12 26 62 : E4
5370 36 68 14 52 14 58 00 1E : 8E
5378 00 2F 2F 2F 2F 2F 2F 2F : 49
----------------------------------
SUM: 8B 46 59 82 7D FA 8D 9B AD9D

5380 2F 2F 02 2F 20 20 20 20 : 0F
5388 20 20 20 2F 02 2F 20 4C : 2C
5390 49 4E 45 52 20 2F 02 2F : AE
5398 20 20 20 20 20 20 20 2F : 0F
53A0 02 2F 20 20 20 20 20 20 : F1
53A8 20 2F 02 2F 2F 2F 2F 2F : 3C
53B0 2F 2F 2F 2F 02 02 2F 2F : 1E
53B8 2F 2F 2F 2F 2F 2F 2F 02 : 4B
53C0 2F 20 52 4F 55 4E 44 20 : F7
53C8 2F 02 2F 20 20 20 20 20 : 00
53D0 20 20 2F 02 2F 2F 2F 2F : 2D
53D8 2F 2F 2F 2F 2F 02 02 2F : 1E
53E0 2F 2F 2F 2F 2F 2F 2F 2F : 78
53E8 02 2F 20 52 29 65 74 72 : 17
53F0 79 2F 02 2F 20 4E 29 65 : D5
53F8 78 74 20 2F 02 2F 20 42 : CE
----------------------------------
SUM: 07 EB 57 FC 2F CE 90 30 1D28

5400 29 61 63 6B 20 2F 02 2F : D8
5408 20 53 29 6B 69 70 20 2F : 2F
5410 02 2F 20 51 29 75 69 74 : 1D
5418 20 2F 02 2F 2F 2F 2F 2F : 3C
5420 2F 2F 2F 2F 00 09 12 43 : 1A
5428 4C 45 41 52 21 00 0D 14 : 66
5430 48 69 74 20 53 50 41 43 : 6C
5438 45 20 4B 65 79 21 00 00 : AF
5440 05 07 16 45 00 0C 07 17 : 91
5448 00 0F 07 17 00 10 09 53 : 99
5450 00 13 07 17 00 16 07 45 : 93
5458 00 16 07 17 00 16 0A 45 : 99
5460 00 16 0D 45 00 1D 07 17 : A3
5468 00 1E 07 42 51 11 91 82 : DC
5470 42 51 13 FF 19 00 00 00 : BE
5478 29 00 00 00 2D 00 00 00 : 56
----------------------------------
SUM: E3 D3 2F 6C 65 33 D3 28 D804

5480 00 00 00 00 00 00 00 00 : 00
5488 25 00 00 00 51 00 00 00 : 76
5490 00 00 00 00 00 00 00 00 : 00
5498 15 00 00 00 09 00 00 00 : 1E
54A0 43 00 00 00 00 00 00 00 : 43
54A8 00 00 00 00 21 00 00 00 : 21
54B0 4D 00 00 00 00 00 00 00 : 4D
54B8 00 00 00 00 01 00 00 00 : 01
54C0 05 00 00 00 3F 00 00 51 : 95
54C8 00 00 00 1B 00 00 00 00 : 1B
54D0 00 00 14 00 00 00 00 3B : 4F
54D8 00 00 00 00 00 00 00 01 : 01
54E0 00 00 00 21 00 27 00 00 : 48
54E8 00 00 42 00 00 00 00 00 : 42
54F0 0E 00 00 00 00 23 00 2B : 5C
54F8 00 00 00 37 00 00 00 00 : 37
----------------------------------
SUM: DD 00 56 73 BB 4A 00 B8 D5B8

5500 00 00 00 05 00 00 00 00 : 05
5508 48 00 00 00 00 00 00 33 : 7B
5510 00 00 00 09 00 00 00 00 : 09
5518 00 00 00 00 00 00 00 00 : 00
5520 00 00 01 00 3D 00 00 00 : 3E
5528 00 10 00 00 00 00 00 34 : 44
5530 00 00 00 00 00 00 00 00 : 00
5538 00 00 00 00 51 00 00 00 : 51
5540 2F 00 00 00 00 00 00 00 : 2F
5548 00 00 00 00 00 0A 00 00 : 0A
5550 00 00 00 26 00 00 00 00 : 26
5558 07 00 43 00 00 00 00 00 : 4A
5560 00 00 00 00 00 00 00 0D : 0D
5568 00 00 00 19 00 00 00 1D : 36
5570 00 00 00 00 00 21 00 00 : 21
5578 00 00 00 05 00 00 00 23 : 28
----------------------------------
SUM: 7E 10 44 52 8E 2B 00 B4 7711

5580 00 00 00 01 00 13 00 29 : 3D
5588 00 00 00 09 00 00 00 15 : 1E
5590 00 00 00 49 00 00 00 2D : 76
5598 00 2B 00 51 00 00 00 3D : B9
55A0 00 00 00 4D 00 00 00 00 : 4D
55A8 00 39 00 00 00 00 00 33 : 6C
55B0 00 00 00 37 00 00 00 45 : 7C
55B8 01 00 00 00 05 00 00 00 : 06
55C0 1B 00 00 00 00 00 00 00 : 1B
55C8 00 00 00 00 00 14 00 00 : 14
55D0 00 00 00 00 00 00 00 00 : 00
55D8 00 20 00 00 0D 00 00 00 : 2D
55E0 3D 00 00 00 4D 00 00 2E : B8
55E8 00 00 00 00 00 00 00 00 : 00
55F0 00 00 00 26 00 00 00 00 : 26
55F8 00 00 00 00 00 00 00 00 : 00
----------------------------------
SUM: 59 84 00 4E 5F 27 00 4E B954

5600 35 00 00 00 41 00 00 00 : 76
5608 51 00 00 00 00 00 00 00 : 51
5610 00 00 00 35 00 2B 00 4F : AF
5618 00 51 00 00 00 00 00 00 : 51
5620 00 00 00 00 00 33 00 00 : 33
5628 00 00 00 45 00 00 00 00 : 45
5630 00 00 00 00 00 00 00 03 : 03
5638 00 00 00 00 00 43 00 00 : 43
5640 00 00 00 00 00 00 00 00 : 00
5648 00 01 00 3D 00 0F 00 13 : 60
5650 00 00 00 00 00 00 00 00 : 00
5658 00 00 00 00 00 2C 00 00 : 2C
5660 00 00 00 00 01 00 00 00 : 01
5668 00 00 51 00 00 00 00 00 : 51
5670 00 00 00 00 00 00 00 00 : 00
5678 2F 00 21 00 00 4C 00 00 : 9C
----------------------------------
SUM: B5 52 72 B7 42 28 00 65 DE31

5680 00 00 00 00 00 00 00 3C : 3C
5688 00 00 17 00 19 00 00 00 : 30
5690 00 00 00 00 00 00 00 00 : 00
5698 00 00 07 00 00 00 00 00 : 07
56A0 0D 00 00 00 00 00 00 44 : 51
56A8 00 00 00 00 00 47 00 00 : 47
56B0 00 51 00 00 00 00 00 00 : 51
56B8 00 00 00 00 00 43 00 3F : 82
56C0 00 00 00 4F 00 0D 00 00 : 5C
56C8 00 00 00 00 00 00 00 00 : 00
56D0 00 00 00 17 00 00 00 00 : 17
56D8 00 00 00 00 00 00 00 00 : 00
56E0 00 39 00 37 00 00 00 27 : 97
56E8 00 01 00 00 00 00 00 00 : 01
56F0 00 00 00 00 00 2F 00 00 : 2F
56F8 00 05 00 00 00 00 00 00 : 05
----------------------------------
SUM: 0D 90 1E 9D 19 C6 00 E6 ED95

5700 00 00 00 00 00 00 01 00 : 01
5708 00 00 00 00 25 00 00 00 : 25
5710 15 00 00 00 19 00 00 00 : 2E
5718 00 00 00 00 00 00 00 00 : 00
5720 00 00 00 00 45 00 00 00 : 45
5728 00 00 00 00 00 00 00 00 : 00
5730 00 00 00 00 07 00 00 00 : 07
5738 4D 00 00 00 33 00 00 00 : 80
5740 00 00 51 00 00 00 00 00 : 51
5748 00 00 00 00 00 00 00 00 : 00
5750 00 00 00 00 00 00 00 00 : 00
5758 00 15 00 00 00 09 00 00 : 1E
5760 00 00 00 11 00 07 00 00 : 18
5768 00 1D 00 00 00 01 00 00 : 1E
5770 00 00 00 3B 00 3D 00 47 : BF
5778 00 00 00 00 00 37 00 00 : 37
----------------------------------
SUM: 62 32 51 4C BD 85 01 47 C142
```

```
5780 00 51 00 00 00 29 00 35 : AF
5788 00 00 00 00 00 2D 00 00 : 2D
5790 00 41 00 00 00 00 00 00 : 41
5798 00 00 00 00 00 00 31 00 : 31
57A0 00 00 00 00 00 00 29 00 : 29
57A8 00 00 00 00 00 00 00 00 : 00
57B0 00 00 51 4C 00 14 0F 00 : C0
57B8 00 00 00 00 00 00 00 00 : 00
57C0 00 00 00 00 00 4E 00 12 : 60
57C8 00 00 00 00 00 00 00 00 : 00
57D0 00 00 00 00 00 00 3D 42 : 7F
57D8 00 06 01 00 00 00 00 00 : 07
57E0 00 00 00 00 00 00 39 00 : 39
57E8 00 00 00 00 00 00 21 00 : 21
57F0 00 00 00 00 00 00 00 00 : 00
57F8 00 00 00 00 3E 00 00 00 : 3E
-----------------------------------
SUM: 00 98 52 4C 3E B8 00 89 09B6

5800 00 00 00 51 00 00 00 19 : 6A
5808 00 00 00 00 00 00 00 00 : 00
5810 00 00 00 00 48 00 34 00 : 7C
5818 3A 00 16 00 00 00 00 00 : 50
5820 00 00 00 00 00 00 00 4D : 4D
5828 00 00 00 01 00 00 00 00 : 01
5830 00 00 08 00 00 00 00 00 : 08
5838 00 00 00 00 00 00 00 00 : 00
5840 51 00 00 00 00 00 00 00 : 51
5848 49 00 00 00 00 00 00 16 : 5F
5850 00 00 00 2A 00 00 05 00 : 2F
5858 00 00 00 00 00 00 00 00 : 00
5860 00 00 00 00 00 00 01 00 : 01
5868 00 00 13 00 00 00 00 00 : 13
5870 00 00 00 00 00 00 00 00 : 00
5878 00 00 1F 00 00 1C 00 00 : 3B
-----------------------------------
SUM: D4 00 50 7C 48 1C 3A 7C 5509

5880 00 22 00 00 00 00 00 00 : 22
5888 35 00 00 00 00 00 00 00 : 35
5890 41 00 00 00 00 01 00 00 : 42
5898 00 00 00 00 00 00 00 00 : 00
58A0 00 00 00 00 00 21 00 00 : 21
58A8 00 25 00 00 00 19 00 00 : 3E
58B0 00 00 00 09 00 00 00 00 : 09
58B8 00 29 00 00 00 00 00 39 : 62
58C0 00 00 34 00 00 2F 00 00 : 63
58C8 00 00 00 41 00 00 00 00 : 41
58D0 00 00 00 00 00 00 00 00 : 00
58D8 00 51 00 00 00 00 00 00 : 51
58E0 00 49 00 00 00 00 00 00 : 49
58E8 00 00 00 00 00 06 00 00 : 06
58F0 00 0A 00 00 00 00 00 00 : 0A
58F8 00 00 00 00 00 00 00 00 : 00
-----------------------------------
SUM: 76 14 34 4A 00 70 00 39 67BA

5900 00 00 00 00 00 00 00 00 : 00
5908 15 00 01 00 2D 00 00 00 : 43
5910 00 00 00 00 00 00 00 00 : 00
5918 00 00 00 00 00 00 00 00 : 00
5920 00 00 00 4C 00 00 00 3C : 88
5928 00 00 00 00 00 00 51 00 : 51
5930 00 00 00                : 00
-----------------------------------
SUM: 15 00 01 4C 2D 00 51 3C C1AD
```

## リスト2

```
                         1 ;//////////////////////////////////////
                         2 ;
                         3 ;      L I N E R   ver 1.0
                         4 ;
                         5 ;//////////////////////////////////////
                         6
5000                     7          ORG     $5000
                         8
(1FFA)                   9 #HOT     EQU     $1FFA
(1FF4)                  10 #PRINT   EQU     $1FF4
(1FF1)                  11 #PRNTS   EQU     $1FF1
(1FEE)                  12 #LTNL    EQU     $1FEE
(1FE5)                  13 #MSX     EQU     $1FE5
(1FE2)                  14 #MPRNT   EQU     $1FE2
(1FDF)                  15 #TAB     EQU     $1FDF
(1FD0)                  16 #GETKY   EQU     $1FD0
(201E)                  17 #LOC     EQU     $201E
(2021)                  18 #FLGET   EQU     $2021
(2030)                  19 #WIDCH`  EQU     $2030
                        20
(4000)                  21 STACK    EQU     $4000
                        22
(4000)                  23 BUF1     EQU     STACK
(4100)                  24 BUF2     EQU     BUF1  +$100
(4200)                  25 POS      EQU     BUF2  +$100
(4202)                  26 DIR      EQU     POS   +2
(4204)                  27 COUNT    EQU     DIR   +2
(4205)                  28 CLRFLG   EQU     COUNT +1
(4206)                  29 DEMFLG   EQU     CLRFLG+1
(4207)                  30 STAGE    EQU     DEMFLG+1
(4208)                  31 STBAK   .EQU     STAGE +1
(4209)                  32 KEYAD    EQU     STBAK +1
(420B)                  33 KEYBUF   EQU     KEYAD +2
                        34
                        35          ;--------------------
                        36
5000  C3 5014           37          JP START
                        38
5003  0F                39 MAX:     DB 15
                        40
5004  2E 2E 00          41 CHR1:    DB "..",0
5007  7B 7B 00          42 CHR2:    DB "{{",0
500A  7B 7B 00          43 CHR3:    DB "{{",0
500D  7B 00             44 CHR4:    DB "{" ,0
500F  20 20 00          45 CHR5:    DB "  ",0
5012  20 00             46 CHR6:    DB " " ,0
                        47
                        48          ;--------------------
                        49 START:
5014  31 4000           50          LD SP,STACK
5017  CD 5302           51          CALL INIT
                        52 START2:
501A  CD 527D           53          CALL TITLE
501D  3E                54          DB [LD A]
                        55 START3:
501E  AF 32 4206        56          XOR A :LD (DEMFLG),A
5022  3E 01 32 4207     57          LD A,1:LD (STAGE),A
5027  CD 5264           58          CALL SCREEN
                        59 RETRY:
502A  31 4000           60          LD SP,STACK
502D  CD 520E           61          CALL ROUND
                        62          [
5030  CD 5064           63            CALL MAINSUB
5033  3A 4205           64            LD A,(CLRFLG)
5036  B7 28 F7          65          ] IF A=0 JR
                        66 CLEAR:
5039  CD 512D           67          CALL SETCSR
503C  11 5007 CD 1FE5   68          LD DE,CHR2  :CALL #MSX
5042  11 5425 CD 51BA   69          LD DE,CLRMSG:CALL PRTMSG
5048  3A 4206           70          LD A,(DEMFLG)
504B  B7 28 05 CD 52C0  71          IF A<>0 THEN CALL TITLE2:JR START2
5051  18 C7
                        72          [
5053  CD 1FD0           73            CALL #GETKY
5056  FE 1B CA 5186     74            IF A=$1B  JP QUIT
505B  FE 20 20 F4       75          ] IF A<>$20 JR
505F  CD 5144           76          CALL NEXT
5062  18 B6             77          JR START2
                        78          ;--------------------
                        79 MAINSUB:
5064  CD 512D           80          CALL SETCSR
5067  2A 4200 24 36 FF  81          LD HL,(POS):INC H:LD (HL),-1
506D  3A 4204 CD 51DD   82          LD A,(COUNT):CALL PRTPCE2
5073  21 0322 CD 201E   83          LD HL,$0322:CALL #LOC
5079  CD 5194           84          CALL INKEY1
507C  CD 508F 30 1D     85          CALL MOVKEY?:JR NC,MOVE
5081  4F                86          LD C,A
                        87          [
5082  7E 23 B7 C8       88            LD A,(HL+):IF A=0 RET
5086  5E 23 56 23       89            LD E,(HL+):LD D,(HL+)
508A  B9 20 F5          90          ] IF A<>C JR
508D  EB                91          EX DE,HL
508E  E9                92          JP (HL)
                        93          ;--------------------
                        94 MOVKEY?:
508F  21 5329 01 000C   95          LD HL,KEYDAT:LD BC,12
5095  ED B1 37 C0       96          CPIR:SCF:RET NZ
5099  79 E6 03 4F       97          LD A,C:AND $03:LD C,A
509D  C9                98          RET
                        99          ;--------------------
                       100 MOVE:
509E  ED 43 4202       101          LD (DIR),BC
50A2  21 5348          102          LD HL,VECTOR
50A5  09 09            103          ADD HL,BC:ADD HL,BC
50A7  4E 23 46         104          LD C,(HL):LD B,(+HL)
50AA  2A 4200 09       105          LD HL,(POS):ADD HL,BC
50AE  11 4204          106          LD DE,COUNT
50B1  7E B7 28 0E      107          LD A,(HL):IF A=0 JR MOVE2
50B5  E6 7F 4F         108          AND $7F:LD C,A
50B8  1A               109          LD A,(DE)
50B9  3C B9 28 0C      110          INC A:IF A=C JR MOVE3
50BD  D6 02 B9 28 29   111          SUB 2:IF A=C JR DELETE
50C2  C9               112          RET
                       113 MOVE2:
50C3  1A C6 81         114          LD A,(DE):ADD A,$81
50C6  77 D6 80         115          LD (HL),A:SUB $80
                       116 MOVE3:
50C9  12               117          LD (DE),A
50CA  E5               118          PUSH HL
50CB  CD 512D          119            CALL SETCSR
50CE  E1               120          POP HL
50CF  22 4200          121          LD (POS),HL
50D2  11 5007 CD 1FE5  122          LD DE,CHR2:CALL #MSX
50D8  11 500A 21 500D  123          LD DE,CHR3:LD HL,CHR4
50DE  CD 5112          124          CALL PRTJNT
50E1  3A 4204 FE 51 C0 125          LD A,(COUNT):IF A<>81 RET
50E7  32 4205          126          LD (CLRFLG),A
50EA  C9               127          RET
                       128          ;--------------------
                       129 DELETE:
50EB  24 7E 25         130          INC H:LD A,(HL):DEC H
50EE  B7 C8            131          IF A=0 RET
50F0  1A 3D 12         132          LD A,(DE):DEC A:LD (DE),A
50F3  E5               133          PUSH HL
50F4  CD 512D          134            CALL SETCSR
50F7  2A 4200          135            LD HL,(POS)
50FA  24 36 00 25      136            INC H:LD (HL),0:DEC H
50FE  7E               137            LD A,(HL)
50FF  CB 7F 28 02 AF 77 138           IF [BIT 7,A]<>0 THEN XOR A:LD (HL),A
5105  CD 51D4          139            CALL PRTPCE1
5108  E1               140          POP HL
5109  22 4200          141          LD (POS),HL
510C  11 500F 21 5012  142          LD DE,CHR5:LD HL,CHR6
                       143 PRTJNT:
5112  D5 E5            144          PUSH DE,HL
5114  21 5350 ED 4B    145            LD HL,CSRDAT:LD BC,(DIR)
5119  4202
511B  09 09            146            ADD HL,BC:ADD HL,BC
511D  09 09            147            ADD HL,BC:ADD HL,BC
511F  EB CD 1FE5       148            EX DE,HL:CALL #MSX
5123  E1 D1            149          POP HL,DE
5125  CB 41 20 01 EB   150          IF [BIT 0,C]=0 THEN EX DE,HL
512A  C3 1FE5          151          JP #MSX
                       152          ;--------------------
                       153 SETCSR:
512D  3A 4200 26 FD    154          LD A,(POS):LD H,-3
5132  24 24 D6 0A 30 FA 155         [ INC H,H:SUB 10 ] JR NC
5138  C6 0A 6F         156          ADD A,10:LD L,A
513B  87 85 6F         157          ADD A,A:ADD A,L:LD L,A
513E  C3 201E          158          JP #LOC
                       159          ;--------------------
                       160 BACK:
5141  3E FF 21         161          LD A,-1:DB [LD HL]
                       162 NEXT:
5144  3E 01            163          LD A,1
5146  CD 5178 D8       164          CALL CHGSTG:RET C
514A  C3 502A          165          JP RETRY
                       166          ;--------------------
                       167 SKIP:
514D  3A 4207 32 4208  168          LD A,(STAGE):LD (STBAK),A
                       169 SKP1:
5153  CD 51E2 CD 519A  170          CALL PRTSTG :CALL INKEY2
5159  CD 508F 38 08    171          CALL MOVKEY?:JR C,SKP2
515E  E6 02 3D CD 5178 172          AND $02:DEC A:CALL CHGSTG
5164  18 ED            173          JR SKP1
                       174 SKP2:
5166  FE 0D CA 502A    175          IF A=$0D  JP RETRY
516B  FE 1B 20 E4      176          IF A<>$1B JR SKP1
516F  3A 4208 32 4207  177          LD A,(STBAK):LD (STAGE),A
5175  C3 51E2          178          JP PRTSTG
                       179          ;--------------------
                       180 CHGSTG:
5178  21 4207          181          LD HL,STAGE
517B  86 37 C8         182          ADD A,(HL):SCF:RET Z
517E  4F 3A 5003       183          LD C,A:LD A,(MAX)
5182  B9 D8            184          CP C:RET C
5184  71               185          LD (HL),C
5185  C9               186          RET
                       187          ;--------------------
                       188 QUIT:
5186  CD 1FE2          189          CALL #MPRNT
5189  0C               190           DB $0C
518A  51 75 69 74 0D 0D 191          DB "Quit",$0D,$0D,0
5190  00
5191  C3 1FFA          192          JP #HOT
                       193          ;--------------------
                       194 INKEY1:
5194  3A 4206 B7 20 0C 195          LD A,(DEMFLG):IF A<>0 JR DINKEY
```

▶アンケートハガキに何か書いて運がよけりゃ誌上に載っけてもらえるんでしょうか？
だったら「求む！　恋人またはおむこさん」とか。　　碓井　理恵(24)和歌山県

```
                        196 INKEY2:
519A  CD 2021           197         CALL #FLGET
519D  FE 61 D8          198         IF A<'a'    RET
51A0  FE 7B D0          199         IF A>='z'+1 RET
51A3  D6 20             200         SUB $20
51A5  C9                201         RET
                        202 DINKEY:
51A6  21 8000           203         LD HL,$8000
51A9  2B 7C B5 20 FB    204         [ DEC HL ] IF HL<>0 JR
51AE  CD 52F4           205         CALL KEYCHK
51B1  2A 4209           206         LD HL,(KEYAD)
51B4  7E 23             207         LD A,(HL+)
51B6  22 4209           208         LD (KEYAD),HL
51B9  C9                209         RET
                        210         ;--------------------
                        211 PRTMSG:
51BA  EB                212         EX DE,HL
51BB  5E 23 56 23       213           LD E,(HL+):LD D,(HL+)
51BF  EB                214         EX DE,HL
51C0  FE                215         DB [CP]
                        216 PRTM1:
51C1  24 CD 201E        217         INC H:CALL #LOC
                        218         [
51C5  1A 13             219           LD A,(DE+)
51C7  B7 C8             220           IF A=0 RET
51C9  FE 02 38 ED       221           CP 2:JR C,PRTMSG
51CD  28 F2             222                JR Z,PRTM1
51CF  CD 1FF4           223           CALL #PRINT
51D2  18 F1             224         ] JR
                        225
                        226 PRTPCE1:
51D4  B7 20 06 11 5004  227         IF A=0 THEN LD DE,CHR1:JP #MSX
51DA  C3 1FE5
                        228 PRTPCE2:
51DD  57 0E 30          229         LD D,A:LD C,'0'
51E0  18 11             230         JR PRTDCM2
                        231
                        232 PRTSTG:
51E2  21 0922 CD 201E   233         LD HL,$0922:CALL #LOC
51E8  3A 4207           234         LD A,(STAGE)
51EB  57 0E 20          235         LD D,A:LD C,' '
51EE  1E 64 CD 51FB     236         LD E,100:CALL PRTD1
                        237 PRTDCM2:
51F3  1E 0A CD 51FB     238         LD E,10:CALL PRTD1
51F8  42                239         LD B,D
51F9  18 0C             240         JR PRTD3
                        241 PRTD1:
51FB  06 00 7A          242         LD B,0:LD A,D
51FE  04 93 30 FC       243         [ INC B:SUB E ] JR NC
5202  83 57             244         ADD A,E:LD D,A
5204  05 28 02          245         DEC B:JR Z,PRTD4
                        246 PRTD3:
5207  0E 30             247         LD C,'0'
                        248 PRTD4:
5209  78 81             249         LD A,B:ADD A,C
520B  C3 1FF4           250         JP #PRINT
                        251         ;--------------------
                        252 ROUND:
520E  CD 51E2           253         CALL PRTSTG
5211  AF 32 4205        254         XOR A:LD (CLRFLG),A
5215  3C 32 4204        255         INC A:LD (COUNT),A
                        256
5219  21 0100 CD 201E   257         LD HL,$0100:CALL #LOC
521F  2A 4207 26 00     258         LD HL,(STAGE):LD H,0
5224  2B 54 5D          259         DEC HL:LD DE,HL
5227  29 29             260         ADD HL,HL:ADD HL,HL
5229  19 29             261         ADD HL,DE:ADD HL,HL
522B  29 29             262         ADD HL,HL:ADD HL,HL
522D  29 19             263         ADD HL,HL:ADD HL,DE
522F  11 5474           264         LD DE,RDATA
5232  19 EB             265         ADD HL,DE:EX DE,HL
5234  21 400A           266         LD HL,BUF1+10
5237  0E 09             267         LD C,9
                        268         [
5239  06 09             269           LD B,9
                        270           [
523B  24 36 00 25       271             INC H:LD (HL),0:DEC H
523F  1A 77             272             LD A,(DE):LD (HL),A
5241  FE 01 20 03 22    273             IF A=1 THEN LD (POS),HL
5246  4200
5248  D9 CD 51D4 D9     274             EXX:CALL PRTPCE1:EXX
524D  CD 1FF1 13 23     275             CALL #PRNTS:INC DE,HL
5252  10 E7             276           ] DJNZ
5254  CD 1FEE 23        277           CALL #LTNL:INC HL
5258  06 1B CD 1FDF CD  278           LD B,27:CALL #TAB:CALL #LTNL
525E  1FEE
5260  0D 20 D6          279         ] IF [DEC C]<>0 JR
5263  C9                280         RET
                        281         ;--------------------
                        282 SCREEN:
5264  3E 0C CD 1FF4     283         LD A,$0C:CALL #PRINT
5269  11 5377 CD 51BA   284         LD DE,SCRMSG:CALL PRTMSG
526F  21 4000 36 FF     285         LD HL,BUF1:LD (HL),$FF
5274  11 4001           286         LD DE,BUF1+1
5277  01 00FF ED B0     287         LD BC,$100-1:LDIR
527C  C9                288         RET
                        289         ;--------------------
                        290 TITLE:
527D  3E 0C CD 1FF4     291         LD A,$0C:CALL #PRINT
5282  CD 1FD0 B7 20 FA  292         [ CALL #GETKY ] IF A<>0 JR
5288  21 420B 22 4209   293         LD HL,KEYBUF:LD (KEYAD),HL
528E  11 542E CD 51BA   294         LD DE,HITMSG:CALL PRTMSG
5294  11 543F           295         LD DE,TTLDAT
                        296         [
5297  CD 52E5 38 24     297           CALL GETD:JR C,TITLE2
529C  D5                298           PUSH DE
                        299           [
529D  CD 52F4           300             CALL KEYCHK
52A0  C5                301             PUSH BC
52A1  CD 201E           302               CALL #LOC
52A4  CD 1FE2 6F 1D 00  303               CALL #MPRNT:DB $6F,$1D,0
52AA  11 2000           304               LD DE,$2000
52AD  1B 7A B3 20 FB    305               [ DEC DE ] IF DE<>0 JR
52B2  3E 7B CD 1FF4     306               LD A,$7B:CALL #PRINT
52B7  CD 52CC           307               CALL MKMOVE
52BA  C1                308             POP BC
52BB  10 E0             309           ] DJNZ
52BD  D1                310           POP DE
52BE  18 D7             311         ] JR
                        312 TITLE2:
52C0  01 3000           313         LD BC,$3000
                        314         [
52C3  CD 52F4 0B        315           CALL KEYCHK:DEC BC
52C7  78 B1 20 F8       316         ] IF BC<>0 JR
52CB  C9                317         RET
                        318
                        319 MKMOVE:
52CC  3E 2D CD 52D3     320         LD A,[DEC L]:CALL MK1
52D1  3E 25             321         LD A,[DEC H]
                        322 MK1:
52D3  CD 52D7 3D        323         CALL MK2:DEC A
                        324 MK2:
52D7  CB 11 D0          325         RL C:RET NC
52DA  32 52DD           326         LD (MK3),A
52DD  00                327 MK3:    NOP
52DE  C9                328         RET
```

```
                        329 GET1:
52DF  EB                330         EX DE,HL
52E0  5E 23 56 23       331         LD E,(HL+):LD D,(HL+)
52E4  EB                332         EX DE,HL
                        333 GETD:
52E5  1A 13             334         LD A,(DE+)
52E7  3C 37 C8          335         INC A:SCF:RET Z
52EA  3D 28 F2          336         DEC A:JR Z,GET1
52ED  4F                337         LD C,A
52EE  E6 0F 47          338         AND $0F:LD B,A
52F1  A9 4F             339         XOR C  :LD C,A
52F3  C9                340         RET
                        341         ;--------------------
                        342 KEYCHK:
52F4  CD 1FD0           343         CALL #GETKY
52F7  FE 1B CA 5186     344         IF A=$1B JP QUIT
52FC  FE 20 CA 501E     345         IF A=$20 JP START3
5301  C9                346         RET
                        347         ;--------------------
                        348 INIT:
5302  3E 0C CD 1FF4     349         LD A,$0C:CALL #PRINT
5307  3E 28 CD 2030     350         LD A,40 :CALL #WIDCH
                        351
530C  11 5360 21 420B   352         LD DE,DEMKEY:LD HL,KEYBUF
                        353         [
5312  1A 13 B7 C8       354           LD A,(DE+):IF A=0 RET
5316  47 E6 0F 4F A8    355           LD B,A:AND $0F:LD C,A:XOR B
531B  0F 0F 0F 0F 47    356           RRCA:RRCA:RRCA:RRCA:LD B,A
5320  79 C6 30          357           LD A,C:ADD A,'0'
5323  77 23 10 FC       358           [ LD (HL+),A ] DJNZ
5327  18 E9             359         ] JR
                        360         ;/////////////////////////////
                        361 KEYDAT:
5329  32 36 38 34 4B 4C 362         DB "2684KLIJ"
532F  49 4A
5331  1F 1C 1E 1D       363         DB $1F,$1C,$1E,$1D
                        364
5335  52 502A           365         DB 'R' :DW RETRY
5338  4E 5144           366         DB 'N' :DW NEXT
533B  42 5141           367         DB 'B' :DW BACK
533E  53 514D           368         DB 'S' :DW SKIP
5341  51 5186           369         DB 'Q' :DW QUIT
5344  1B 5186           370         DB $1B :DW QUIT
5347  00                371         DB 0
                        372 VECTOR:
5348  FFFF FFF6 0001    373         DW -1,-10,1,10
534E  000A
                        374 CSRDAT:
5350  1D 1D 1D 00       375         DH 1D 1D 1D 00
5354  1E 1D 1D 00       376         DH 1E 1D 1D 00
5358  00 00 00 00       377         DH 00 00 00 00
535C  1F 1D 1D 00       378         DH 1F 1D 1D 00
                        379 DEMKEY:
5360  46 58 14 42 34 78 380         DH 46 58 14 42 34 78 16 62 16 68 66 12
5366  16 62 16 68 66 12
536C  54 12 26 62 36 68 381         DH 54 12 26 62 36 68 14 52 14 58 00
5372  14 52 14 58 00
                        382         ;--------------------
                        383 SCRMSG:
5377  1E 00             384         DB 30,0
5379  2F 2F 2F 2F 2F 2F 385         DB "/////////",2
537F  2F 2F 2F 02
5383  2F 20 20 20 20 20 386         DB "/       /",2
5389  20 20 2F 02
538D  2F 20 4C 49 4E 45 387         DB "/ LINER /",2
5393  52 20 2F 02
5397  2F 20 20 20 20 20 388         DB "/       /",2
539D  20 20 2F 02
53A1  2F 20 20 20 20 20 389         DB "/       /",2
53A7  20 20 2F 02
53AB  2F 2F 2F 2F 2F 2F 390         DB "/////////",2,2
53B1  2F 2F 2F 02 02
53B6  2F 2F 2F 2F 2F 2F 391         DB "/////////",2
53BC  2F 2F 2F 02
53C0  2F 20 52 4F 55 4E 392         DB "/ ROUND /",2
53C6  44 20 2F 02
53CA  2F 20 20 20 20 20 393         DB "/       /",2
53D0  20 20 2F 02
53D4  2F 2F 2F 2F 2F 2F 394         DB "/////////",2,2
53DA  2F 2F 2F 02 02
53DF  2F 2F 2F 2F 2F 2F 395         DB "/////////",2
53E5  2F 2F 2F 02
53E9  2F 20 52 29 65 74 396         DB "/ R)etry/",2
53EF  72 79 2F 02
53F3  2F 20 4E 29 65 78 397         DB "/ N)ext /",2
53F9  74 20 2F 02
53FD  2F 20 42 29 61 63 398         DB "/ B)ack /",2
5403  6B 20 2F 02
5407  2F 20 53 29 6B 69 399         DB "/ S)kip /",2
540D  70 20 2F 02
5411  2F 20 51 29 75 69 400         DB "/ Q)uit /",2
5417  74 20 2F 02
541B  2F 2F 2F 2F 2F 2F 401         DB "/////////",0
5421  2F 2F 2F 00
                        402
5425  09 12 43 4C 45 41 403 CLRMSG: DB  9,18,"CLEAR!",0
542B  52 21 00
542E  0D 14 48 69 74 20 404 HITMSG: DB 13,20,"Hit SPACE Key!",0
5434  53 50 41 43 45 20
543A  4B 65 79 21 00
                        405         ;--------------------
                        406 TTLDAT:
543F  00 05 07 16 45    407         DB 0, 5,7,$16,$45                ;L
5444  00 0C 07 17       408         DB 0,12,7,$17                    ;I
5448  00 0F 07 17 00 10 409         DB 0,15,7,$17, 0,16,9,$53        ;N
544E  09 53
5450  00 13 07 17       410         DB   0,19,7,$17
5454  00 16 07 45 00 16 411         DB 0,22,7,$45, 0,22,7,$17        ;E
545A  07 17
545C  00 16 0A 45 00 16 412         DB   0,22,10,$45, 0,22,13,$45
5462  0D 45
5464  00 1D 07 17 00 1E 413         DB 0,29,7,$17, 0,30,7,$42,$51 ;R
546A  07 42 51
546D  11 91 82 42 51 13 414         DB   $11,$91,$82,$42,$51,$13
5473  FF                415         DB -1
                        416         ;--------------------
                        417 RDATA:
5474  19 00 00 00 29 00 418         DB 25, 0, 0, 0,41, 0, 0, 0,45 ;1
547A  00 00 2D
547D  00 00 00 00 00 00 419         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5483  00 00 00
5486  00 00 25 00 00 00 420         DB  0, 0,37, 0, 0, 0,81, 0, 0
548C  51 00 00
548F  00 00 00 00 00 00 421         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5495  00 00 00
5498  15 00 00 00 09 00 422         DB 21, 0, 0, 0,09, 0, 0, 0,67
549E  00 00 43
54A1  00 00 00 00 00 00 423         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
54A7  00 00 00
54AA  00 00 21 00 00 00 424         DB  0, 0,33, 0, 0, 0,77, 0, 0
54B0  4D 00 00
54B3  00 00 00 00 00 00 425         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
54B9  00 00 00
54BC  01 00 00 00 05 00 426         DB 01, 0, 0, 0,05, 0, 0, 0,63
54C2  00 00 3F
                        427
```

▶「F-Card GT」が面白そうですね。なかなか便利そうなので使ってみようかと考えています。
渡辺　雅孝(32)栃木県

```
54C5  00 00 51 00 00 00    428         DB  0, 0,81, 0, 0, 0,27, 0, 0 ;2
54CB  1B 00 00
54CE  00 00 00 00 14 00    429         DB  0, 0, 0, 0,20, 0, 0, 0, 0
54D4  00 00 00
54D7  3B 00 00 00 00 00    430         DB 59, 0, 0, 0, 0, 0, 0, 0,01
54DD  00 00 01
54E0  00 00 00 21 00 27    431         DB  0, 0, 0,33, 0,39, 0, 0, 0
54E6  00 00 00
54E9  00 42 00 00 00 00    432         DB  0,66, 0, 0, 0, 0, 0,14, 0
54EF  00 0E 00
54F2  00 00 00 23 00 2B    433         DB  0, 0, 0,35, 0,43, 0, 0, 0
54F8  00 00 00
54FB  37 00 00 00 00 00    434         DB 55, 0, 0, 0, 0, 0, 0, 0,05
5501  00 00 05
5504  00 00 00 00 48 00    435         DB  0, 0, 0, 0,72, 0, 0, 0, 0
550A  00 00 00
550D  00 00 33 00 00 00    436         DB  0, 0,51, 0, 0, 0,09, 0, 0
5513  09 00 00
                           437
5516  00 00 00 00 00 00    438         DB  0, 0, 0, 0, 0, 0, 0, 0, 0 ;3
551C  00 00 00
55:F  00 00 00 01 00 3D    439         DB  0, 0, 0,01, 0,61, 0, 0, 0
5525  00 00 00
5528  00 10 00 00 00 00    440         DB  0,16, 0, 0, 0, 0, 0,52, 0
552E  00 34 00
5531  00 00 00 00 00 00    441         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5537  00 00 00
553A  00 00 51 00 00 00    442         DB  0, 0,81, 0, 0, 0,47, 0, 0
5540  2F 00 00
5543  00 00 00 00 00 00    443         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5549  00 00 00
554C  00 0A 00 00 00 00    444         DB  0,10, 0, 0, 0, 0, 0,38, 0
5552  00 26 00
5555  00 00 00 07 00 43    445         DB  0, 0, 0,07, 0,67, 0, 0, 0
555B  00 00 00
555E  00 00 00 00 00 00    446         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5564  00 00 00
                           447
5567  0D 00 00 00 19 00    448         DB 13, 0, 0, 0,25, 0, 0, 0,29 ;4
556D  00 00 1D
5570  00 00 00 00 00 21    449         DB  0, 0, 0, 0, 0,33, 0, 0, 0
5576  00 00 00
5579  00 00 05 00 00 00    450         DB  0, 0,05, 0, 0, 0,35, 0, 0
557F  23 00 00
5582  00 01 00 13 00 29    451         DB  0,01, 0,19, 0,41, 0, 0, 0
5588  00 00 00
558B  09 00 00 00 15 00    452         DB 09, 0, 0, 0,21, 0, 0, 0,73
5591  00 00 49
5594  00 00 00 2D 00 2B    453         DB  0, 0, 0,45, 0,43, 0,81, 0
559A  00 51 00
559D  00 00 3D 00 00 00    454         DB  0, 0,61, 0, 0, 0,77, 0, 0
55A3  4D 00 00
55A6  00 00 00 39 00 00    455         DB  0, 0, 0,57, 0, 0, 0, 0, 0
55AC  00 00 00
55AF  33 00 00 00 37 00    456         DB 51, 0, 0, 0,55, 0, 0, 0,69
55B5  00 00 45
                           457
55B8  01 00 00 00 05 00    458         DB 01, 0, 0, 0,05, 0, 0, 0,27 ;5
55BE  00 00 1B
55C1  00 00 00 00 00 00    459         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
55C7  00 00 00
55CA  00 00 00 14 00 00    460         DB  0, 0, 0,20, 0, 0, 0, 0, 0
55D0  00 00 00
55D3  00 00 00 00 00 00    461         DB  0, 0, 0, 0, 0, 0,32, 0, 0
55D9  20 00 00
55DC  0D 00 00 00 3D 00    462         DB 13, 0, 0, 0,61, 0, 0, 0,77
55E2  00 00 4D
55E5  00 00 2E 00 00 00    463         DB  0, 0,46, 0, 0, 0, 0, 0, 0
55EB  00 00 00
55EE  00 00 00 00 00 26    464         DB  0, 0, 0, 0, 0,38, 0, 0, 0
55F4  00 00 00
55F7  00 00 00 00 00 00    465         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
55FD  00 00 00
5600  35 00 00 00 41 00    466         DB 53, 0, 0, 0,65, 0, 0, 0,81
5606  00 00 51
                           467
5609  00 00 00 00 00 00    468         DB  0, 0, 0, 0, 0, 0, 0, 0, 0 ;6
560F  00 00 00
5612  00 35 00 2B 00 4F    469         DB  0,53, 0,43, 0,79, 0,81, 0
5618  00 51 00
561B  00 00 00 00 00 00    470         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5621  00 00 00
5624  00 33 00 00 00 00    471         DB  0,51, 0, 0, 0, 0, 0,69, 0
562A  00 45 00
562D  00 00 00 00 00 00    472         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5633  00 00 00
5636  00 03 00 00 00 00    473         DB  0,03, 0, 0, 0, 0, 0,67, 0
563C  00 43 00
563F  00 00 00 00 00 00    474         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5645  00 00 00
5648  00 01 00 3D 00 0F    475         DB  0,01, 0,61, 0,15, 0,19, 0
564E  00 13 00
5651  00 00 00 00 00 00    476         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5657  00 00 00
                           477
565A  00 00 00 2C 00 00    478         DB  0, 0, 0,44, 0, 0, 0, 0, 0 ;7
5660  00 00 00
5663  00 01 00 00 00 00    479         DB  0,01, 0, 0, 0, 0, 0,81, 0
5669  00 51 00
566C  00 00 00 00 00 00    480         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5672  00 00 00
5675  00 00 00 2F 00 21    481         DB  0, 0, 0,47, 0,33, 0, 0,76
567B  00 00 4C
567E  00 00 00 00 00 00    482         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5684  00 00 00
5687  3C 00 00 17 00 19    483         DB 60, 0, 0,23, 0,25, 0, 0, 0
568D  00 00 00
5690  00 00 00 00 00 00    484         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5696  00 00 00
5699  00 07 00 00 00 00    485         DB  0,07, 0, 0, 0, 0, 0,13, 0
569F  00 0D 00
56A2  00 00 00 00 00 44    486         DB  0, 0, 0, 0, 0,68, 0, 0, 0
56A8  00 00 00
                           487
56AB  00 00 47 00 00 00    488         DB  0, 0,71, 0, 0, 0,81, 0, 0 ;8
56B1  51 00 00
56B4  00 00 00 00 00 00    489         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
56BA  00 00 00
56BD  43 00 3F 00 00 00    490         DB 67, 0,63, 0, 0, 0,79, 0,13
56C3  4F 00 0D
56C6  00 00 00 00 00 00    491         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
56CC  00 00 00
56CF  00 00 00 00 17 00    492         DB  0, 0, 0, 0,23, 0, 0, 0, 0
56D5  00 00 00
56D8  00 00 00 00 00 00    493         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
56DE  00 00 00
56E1  39 00 37 00 00 00    494         DB 57, 0,55, 0, 0, 0,39, 0,01
56E7  27 00 01
56EA  00 00 00 00 00 00    495         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
56F0  00 00 00
56F3  00 00 2F 00 00 00    496         DB  0, 0,47, 0, 0, 0,05, 0, 0
56F9  05 00 00
                           497
56FC  00 00 00 00 00 00    498         DB  0, 0, 0, 0, 0, 0, 0, 0, 0 ;9
5702  00 00 00
5705  00 01 00 00 00 00    499         DB  0,01, 0, 0, 0, 0, 0,37, 0
570B  00 25 00
570E  00 00 15 00 00 00    500         DB  0, 0,21, 0, 0, 0,25, 0, 0
5714  19 00 00
5717  00 00 00 00 00 00    501         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
571D  00 00 00
5720  00 00 00 00 45 00    502         DB  0, 0, 0, 0,69, 0, 0, 0, 0
5726  00 00 00
5729  00 00 00 00 00 00    503         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
572F  00 00 00
5732  00 00 07 00 00 00    504         DB  0, 0,07, 0, 0, 0,77, 0, 0
5738  4D 00 00
573B  00 33 00 00 00 00    505         DB  0,51, 0, 0, 0, 0, 0,81, 0
5741  00 51 00
5744  00 00 00 00 00 00    506         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
574A  00 00 00
                           507
574D  00 00 00 00 00 00    508         DB  0, 0, 0, 0, 0, 0, 0, 0, 0 ;10
5753  00 00 00
5756  00 00 00 15 00 00    509         DB  0, 0, 0,21, 0, 0, 0,09, 0
575C  00 09 00
575F  00 00 00 00 11 00    510         DB  0, 0, 0, 0,17, 0,07, 0, 0
5765  07 00 00
5768  00 1D 00 00 00 01    511         DB  0,29, 0, 0, 0,01, 0, 0, 0
576E  00 00 00
5771  00 00 3B 00 3D 00    512         DB  0, 0,59, 0,61, 0,71, 0, 0
5777  47 00 00
577A  00 00 00 37 00 00    513         DB  0, 0, 0,55, 0, 0, 0,81, 0
5780  00 51 00
5783  00 00 29 00 35 00    514         DB  0, 0,41, 0,53, 0, 0, 0, 0
5789  00 00 00
578C  00 2D 00 00 00 41    515         DB  0,45, 0, 0, 0,65, 0, 0, 0
5792  00 00 00
5795  00 00 00 00 00 00    516         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
579B  00 00 00
                           517
579E  31 00 00 00 00 00    518         DB 49, 0, 0, 0, 0, 0, 0, 0,41 ;11
57A4  00 00 29
57A7  00 00 00 00 00 00    519         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
57AD  00 00 00
57B0  00 00 51 4C 00 14    520         DB  0, 0,81,76, 0,20,15, 0, 0
57B6  0F 00 00
57B9  00 00 00 00 00 00    521         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
57BF  00 00 00
57C2  00 00 00 4E 00 12    522         DB  0, 0, 0,78, 0,18, 0, 0, 0
57C8  00 00 00
57CB  00 00 00 00 00 00    523         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
57D1  00 00 00
57D4  00 00 3D 42 00 06    524         DB  0, 0,61,66, 0,06,01, 0, 0
57DA  01 00 00
57DD  00 00 00 00 00 00    525         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
57E3  00 00 00
57E6  39 00 00 00 00 00    526         DB 57, 0, 0, 0, 0, 0, 0, 0,33
57EC  00 00 21
                           527
57EF  00 00 00 00 00 00    528         DB  0, 0, 0, 0, 0, 0, 0, 0, 0 ;12
57F5  00 00 00
57F8  00 00 00 00 3E 00    529         DB  0, 0, 0, 0,62, 0, 0, 0, 0
57FE  00 00 00
5801  00 00 51 00 00 00    530         DB  0, 0,81, 0, 0, 0,25, 0, 0
5807  19 00 00
580A  00 00 00 00 00 00    531         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5810  00 00 00
5813  00 48 00 34 00 3A    532         DB  0,72, 0,52, 0,58, 0,22, 0
5819  00 16 00
581C  00 00 00 00 00 00    533         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5822  00 00 00
5825  00 00 4D 00 00 00    534         DB  0, 0,77, 0, 0, 0,01, 0, 0
582B  01 00 00
582E  00 00 00 00 08 00    535         DB  0, 0, 0, 0,08, 0, 0, 0, 0
5834  00 00 00
5837  00 00 00 00 00 00    536         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
583D  00 00 00
                           537
5840  51 00 00 00 00 00    538         DB 81, 0, 0, 0, 0, 0, 0, 0,73 ;13
5846  00 00 49
5849  00 00 00 00 00 00    539         DB  0, 0, 0, 0, 0, 0,22, 0, 0
584F  16 00 00
5852  00 2A 00 00 05 00    540         DB  0,42, 0, 0,05, 0, 0, 0, 0
5858  00 00 00
585B  00 00 00 00 00 00    541         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5861  00 00 00
5864  00 00 01 00 00 00    542         DB  0, 0,01, 0, 0, 0,19, 0, 0
586A  13 00 00
586D  00 00 00 00 00 00    543         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5873  00 00 00
5876  00 00 00 00 1F 00    544         DB  0, 0, 0, 0,31, 0, 0,28, 0
587C  00 1C 00
587F  00 00 22 00 00 00    545         DB  0, 0,34, 0, 0, 0, 0, 0, 0
5885  00 00 00
5888  35 00 00 00 00 00    546         DB 53, 0, 0, 0, 0, 0, 0, 0,65
588E  00 00 41
                           547
5891  00 00 00 00 01 00    548         DB  0, 0, 0, 0,01, 0, 0, 0, 0 ;14
5897  00 00 00
589A  00 00 00 00 00 00    549         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
58A0  00 00 00
58A3  00 00 21 00 00 00    550         DB  0, 0,33, 0, 0, 0,37, 0, 0
58A9  25 00 00
58AC  00 19 00 00 00 00    551         DB  0,25, 0, 0, 0, 0, 0,09, 0
58B2  00 09 00
58B5  00 00 00 00 29 00    552         DB  0, 0, 0, 0,41, 0, 0, 0, 0
58BB  00 00 00
58BE  00 39 00 00 34 00    553         DB  0,57, 0, 0,52, 0, 0,47, 0
58C4  00 2F 00
58C7  00 00 00 00 41 00    554         DB  0, 0, 0, 0,65, 0, 0, 0, 0
58CD  00 00 00
58D0  00 00 00 00 00 00    555         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
58D6  00 00 00
58D9  51 00 00 00 00 00    556         DB 81, 0, 0, 0, 0, 0, 0, 0,73
58DF  00 00 49
                           557
58E2  00 00 00 00 00 00    558         DB  0, 0, 0, 0, 0, 0, 0, 0, 0 ;15
58E8  00 00 00
58EB  00 00 06 00 00 00    559         DB  0, 0,06, 0, 0, 0,10, 0, 0
58F1  0A 00 00
58F4  00 00 00 00 00 00    560         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
58FA  00 00 00
58FD  00 00 00 00 00 00    561         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5903  00 00 00
5906  00 00 15 00 01 00    562         DB  0, 0,21, 0,01, 0,45, 0, 0
590C  2D 00 00
590F  00 00 00 00 00 00    563         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
5915  00 00 00
5918  00 00 00 00 00 00    564         DB  0, 0, 0, 0, 0, 0, 0, 0, 0
591E  00 00 00
5921  00 00 4C 00 00 00    565         DB  0, 0,76, 0, 0, 0,60, 0, 0
5927  3C 00 00
592A  00 00 00 00 51 00    566         DB  0, 0, 0, 0,81, 0, 0, 0, 0
5930  00 00 00

   0 Error(s) :    83 Symbol(s)
```

# 全機種共通システムインデックス

■85年6月号
序論　共通化の試み
第1部　S-OS"MACE"
第2部　Lisp-85インタプリタ
第3部　チェックサムプログラム
■85年7月号
第4部　マシン語プログラム開発入門
第5部　エディタアセンブラZEDA
第6部　デバッグツールZAID
■85年8月号
第7部　ゲーム開発パッケージBEMS
第8部　ソースジェネレータZING
■85年9月号
インタラプト　S-OS番外地
第9部　マシン語入力ツールMACINTO-S
第10部　Lisp-85入門(1)
■85年10月号
第11部　仮想マシンCAP-X85
連載　Lisp-85入門(2)
■85年11月号
連載　Lisp-85入門(3)
■85年12月号
第12部　Prolog-85発表
■86年1月号
第13部　リロケータブルのお話
第14部　FM音源サウンドエディタ
■86年2月号
第15部　S-OS"SWORD"
第16部　Prolog-85入門(1)
■86年3月号
第17部　magiFORTH発表
連載　Prolog-85入門(2)
■86年4月号
第18部　思考ゲームJEWEL
第19部　LIFE GAME
連載　基礎からのmagiFORTH
連載　Prolog-85入門(3)
■86年5月号
第20部　スクリーンエディタE-MATE
連載　実戦演習magiFORTH
■86年6月号
第21部　Z80TRACER
第22部　magiFORTH TRACER
第23部　ディスクダンプ&エディタ
第24部　"SWORD" 2000 QD
連載　対話で学ぶ magiFORTH
特別付録　PC-8801版S-OS"SWORD"
■86年7月号
第25部　FM音源ミュージックシステム
付録　FM音源ボードの製作
連載　計算力アップのmagiFORTH
特別付録　SMC-777版 S-OS"SWORD"
■86年8月号
第26部　対局五目並べ
第27部　MZ-2500版 S-OS"SWORD"
■86年9月号
第28部　FuzzyBASIC 発表
連載　明日に向かって magiFORTH
■86年10月号
第29部　ちょっと便利な拡張プログラム
第30部　ディスクモニタ DREAM
第31部　FuzzyBASIC 料理法<1>
■86年11月号
第32部　パズルゲーム HOTTAN
第33部　MAZE in MAZE
連載　FuzzyBASIC 料理法<2>
■86年12月号
第34部　CASL & COMET
連載　FuzzyBASIC 料理法<3>
■87年1月号
第35部　マシン語入力ツールMACINTO-C
連載　FuzzyBASIC 料理法<4>
■87年2月号
第36部　アドベンチャーゲーム MARMALADE
第37部　テキアベ作成ツール CONTEX
■87年3月号
第38部　魔法使いはアニメがお好き
第39部　アニメーションツール MAGE
付録　"SWORD" 再掲載と MAGIC の標準化
■87年4月号
第40部　INVADER GAME
第41部　TANGERINE
■87年5月号
第42部　S-OS"SWORD" 変身セット
第43部　MZ-700用 "SWORD" を QD 対応に
■87年6月号
インタラプト　コンパイラ物語
第44部　FuzzyBASIC コンパイラ
第45部　エディタアセンブラ ZEDA-3
■87年7月号
第46部　STORY MASTER
■87年8月号
第47部　パズルゲーム碁石拾い
第48部　漢字出力パッケージ JACKWRITE
特別付録　FM-7/77版 S-OS"SWORD"
■87年9月号
第49部　リロケータブル逆アセンブラ Inside-R
特別付録　PC-8001/8801 版 S-OS"SWORD"
■87年10月号
第50部　tiny CORE WARS
第51部　FuzzyBASIC コンパイラの拡張
第52部　X1turbo 版 S-OS"SWORD"
■87年11月号
序論　神話のなかのマイクロコンピュータ
付録　S-OS の仲間たち
第53部　もうひとつの FuzzyBASIC 入門
第54部　ファイルアロケータ&ローダ
インタラプト　S-OS こちら集中治療室
第55部　BACK GAMMON
■87年12月号
第56部　タートルグラフィックパッケージTURTLE
第57部　X1turbo 版 "SWORD" アフターケア ラインプリントルーチン
特別付録　PASOPIA7 版 S-OS"SWORD"
■88年1月号
第58部　FuzzyBASIC コンパイラ・奥村版
付録　石上版コンパイラ拡張部の修正
■88年2月号
第59部　シューティングゲーム ELFES
■88年3月号
第60部　構造型コンパイラ言語 SLANG
■88年4月号
第61部　デバッギングツール TRADE
第62部　シミュレーションウォーゲーム WALRUS
■88年5月号
第63部　シューティングゲーム ELFES II
第64部　地底最大の作戦
■88年6月号
第65部　構造化言語 SLANG 入門(1)
第66部　Lisp-85 用 NAMPA シミュレーション
■88年7月号
第67部　マルチウィンドウドライバ MW-1
連載　構造化言語 SLANG 入門(2)
■88年8月号
第68部　マルチウィンドウエディタ WINER
■88年9月号
第69部　超小型エディタ TED-750
第70部　アフターケア WINER の拡張
■88年10月号
第71部　SLANG 用ファイル入出力ライブラリ
第72部　シューティングゲーム MANKAI
■88年11月号
第73部　シューティングゲーム ELFES Ⅳ
■88年12月号
第74部　ソースジェネレータ SOURCERY
■89年1月号
第75部　パズルゲーム LAST ONE
第76部　ブロックゲーム FLICK
■89年2月号
第77部　高速エディタアセンブラ REDA
特別付録　X1版 S-OS"SWORD"<再掲載>
■89年3月号
第78部　Z80用浮動小数点演算パッケージSOROBAN
■89年4月号
第79部　SLANG 用実数演算ライブラリ
■89年5月号
第80部　ソースジェネレータ RING
■89年6月号
第81部　超小型コンパイラTTC
■89年7月号
第82部　TTC用パズルゲーム　TICBAN
■89年8月号
第83部　CP/M用ファイルコンバータ
■89年9月号
第84部　生物進化シミュレーションBUGS
■89年10月号
第85部　小型インタプリタ言語TTI
■89年11月号
第86部　TTI用パズルゲーム　PUSH BON!
■89年12月号
第87部　SLANG用リダイレクションライブラリ DIO. LIB
■90年1月号
第88部　SLANG用ゲームWORM KUN
特別付録　再掲載SLANGコンパイラ
■90年2月号
第89部　超小型コンパイラTTC++
■90年3月号
第90部　超多機能アセンブラOHM-Z80
■90年4月号
第91部　ファジィコンピュータシミュレーションI-MY
■90年5月号
第92部　インタプリタ言語STACK
■90年6月号
第93部　リロケータブルフォーマットの取り決め
第94部　STACK用ゲーム　SQUASH!
第95部　X68000対応S-OS "SWORD"
特別付録　PC-286対応S-OS "SWORD"
■90年7月号
第96部　リロケータブルアセンブラWZD
■90年8月号
第97部　リンカWLK
■90年9月号
第98部　BILLIARDS
■90年10月号
第99部　ライブラリアンWLB
■90年11月号
第100部　タブコード対応エディタEDC-T
■90年12月号
第101部　STACKコンパイラ
■91年1月号
第102部　ブロックアクションゲーム COLUMNS
■91年2月号
第103部　ダイスゲームKISMET
■91年3月号
第104部　アクションゲームMUD BALLIN'
■91年4月号
第105部　SLANG用カードゲームDOBON
■91年5月号
第106部　実数型コンパイラ言語REAL
■91年6月号
第107部　Small-C処理系の移植
■91年7月号
第108部　REALソースリスト編
■91年8月号
第109部　Small-Cライブラリの移植
■91年9月号
第110部　SLANG用NEWファイル出力ライブラリ
■91年10月号
第111部　Small-C活用講座(初級編)
■91年11月号
第112部　Small-C活用講座(応用編)
第113部　MORTAL
■91年12月号
第114部　Small-C SLANGコンパチ関数

＊以上のアプリケーションは，基本システムであるS-OS "MACE" またはS-OS"SWORD" がないと動作しませんのでご注意ください。

INDEX

TIME: 0:00:13

MAGICver.2.0対応ゲーム

# 3D MAZE

Mituishi Kazuhiro 光石 和弘

ようやくMAGICを使用したアプリケーションをお届けできます。具体的な使い方を把握していないユーザーの人は，プログラムを見てどのような使われ方をしているか理解してみてください。そして，面白いアプリケーションを作ろうではありませんか。

## 3D迷路だ

3DグラフィックパッケージMAGICを使った迷路ゲームを発表します。ゲームのルールは簡単，スタート地点から見える赤いゴールに向かって進むだけです。画面写真を見ればわかるとおり，いわゆる3D迷路であるダンジョンタイプの迷路ではなく，遊園地などにある柵で囲まれた迷路タイプになっています。

そして，自分は少し上からの視点で見ていることになっていて，迷路はひとつ先の曲がり角まで表示しています。

MAGICを使っているからには，当然，曲がるときにはきれいな回転をしてくれます。ゲーム自体が地味なので，あまり目立たないかもしれません。しかし，なかなか気持ちいいな，と自分では思っています。迷路表示部分以外（上半分）が何も表示されないので，少しさみしいと思いますがこれはしょうがないのでがまんしてください。視点をもう少し自由に変えられたら，面白いかもしれませんね。

そうそう，迷路が複雑すぎて解けないからといって方眼紙にマッピングなんかしちゃいけませんよ。ぜひ，自分の記憶力のみで解きましょう。

## 入力方法

リスト1のダンプリストをマシン語入力ツール上から打ち込みます。マシン語入力ツールは1991年9月号に掲載されたX-BASIC版を入力するか（できるだけコンパイルして使うこと），1990年6月号の付録ディスクに収録されていた“MAC.X”を使用してください。打ち込み終わったデータをセーブする際は3008バイトを指定します。

なお，ダンプリストはLH.Xで圧縮されています。展開にはLH.XまたはLHA.Xが必要になります。お持ちでない方はOh!Xの過去の付録ディスクまたは電脳倶楽部などで入手可能です。打ち込んだデータをMAZE.LZHでセーブしていたならば，

A>LH −E MAZE

のように操作してください。カレントディレクトリ上に“MAZE.X”という実行ファイルが作成されます。

実行する際には，必ず9月号で掲載された「MAGICver.2.0」を先に組み込んでおいてください。拡張モードを使用しているため，5月号で掲載されたバージョン,1.0では動かすことができません。

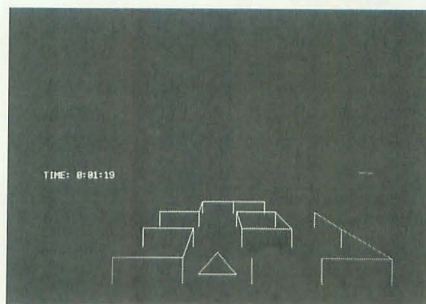
## 操作方法

最初にいっておきますが，ゲームを遊ぶためにはジョイスティックが必要です。

で，ゲームを起動するとタイトルが表示され，トリガAを押すことでゲームが始まります。迷路内の移動とジョイスティックの対応は以下のとおりです。

前進<br>
↑<br>
左回転 ← ○ → 右回転<br>
↓<br>
後退

トリガA：ゲームスタート<br>
トリガB：強制ゲームオーバー（自爆）<br>
BREAKキー：終了

操作についてはこれ以上，説明の必要はないでしょう。

## MAGICver.1.0への対応

入力方法の説明のところで「このゲームは，MAGICver.2.0でしか動作しない」といいました。しかし，9月号を参照してもらえればわかるとおり，拡張モードというのは物体ごとに描画色を設定するデータが3Dデータの中に含まれるだけです。

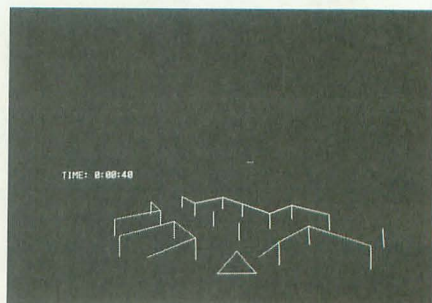
ということは，このプログラム中にある3Dデータの描画色を格納している部分を削ってやればMAGICver.1.0でも大丈夫なはずです。あとは初期化データ部分で描画色の設定，画面モードの設定データを書き換えるだけです。

この場合，ソースリストを打ち込んで改造する必要があります。また，とりあえずダンプリストで打ち込んでおき，disなどを使ってソースリストを生成してから改造を行いましょう。ゴールが見づらくなりますが（同じ色で表示されるため）ゲームを遊ぶことはできます。

いっておきますが，改造は個人の責任において行ってください。

## 最後に

今回のこのプログラムは，アセンブラで開発を始めて間もない私ですからずいぶんと無駄が多いと思います。特に多分岐でテ

ひとつ先の曲がり角まで表示される

表示されている迷路が滑らかに回転

ーブルを使えば，もう少しすっきりとプログラムがまとめられたのではないか，といまさらながらに思っています。さらに，自動迷路生成ルーチンを付ければ，もっと面白くできたかもしれませんね。これは次回への課題かな。

最後に，このプログラムがMAGICを使おうとしているユーザーの皆さんに参考になればとってもうれしいな，などと考えつつ次のプログラムを作ることにしましょう。

では，ぐりんぐりん回転する巨大迷路を抜けるまでがんばってください。

リスト1

```
0000   1C A0 2D 6C 68 31 2D 9F  : BA
0008   0B 00 00 5A 20 00 00 B5  : 3A
0010   7E 30 17 20 00 06 6D 61  : B9
0018   7A 65 2E 78 1E 0C EA 78  : 11
0020   71 92 80 0C AF DB 0E FE  : 25
0028   8B C1 20 0C F4 4A 86 68  : A4
0030   40 1B EE 45 FC A7 F3 39  : 5D
0038   DA ED DA 6A E3 00 67 76  : CB
0040   56 3C D8 58 D2 16 27 3F  : 10
0048   D3 AF 03 D8 63 B9 C3 F4  : 30
0050   C6 F2 E4 28 00 ED 78 61  : 8A
0058   12 E3 E7 CC 5F 80 B9 AF  : EF
0060   6C F4 EE 92 AB 78 16 CF  : E8
0068   B3 8D AF 8B 0F 9F 56 29  : A7
0070   B7 91 AD BB 1D 82 6F 36  : F4
0078   8D 86 EF F3 37 B0 8D 73  : DC
-----------------------------------
SUM:   99 E8 B9 14 CA 94 F5 26  C198

0080   55 53 30 11 44 76 07 3C  : E6
0088   7E 07 40 62 C2 59 AD 94  : 83
0090   28 80 FC 9F 29 05 80 67  : 58
0098   2B E8 D8 29 83 E6 53 F1  : C1
00A0   47 8D AE 6A 7E 37 12 2D  : E0
00A8   66 7B 58 18 B6 21 DB 19  : 1C
00B0   92 F9 E0 2F 11 83 10 38  : 76
00B8   DE 81 C3 34 31 C5 FD 51  : 9A
00C0   30 38 EA 7C 23 04 79 DF  : 4D
00C8   B8 EC 05 33 99 86 C0 F2  : AD
00D0   EB 8A 9C 39 3B 37 C4 7A  : FA
00D8   A3 E0 3B 13 D9 4C 00 BE  : B4
00E0   C4 30 1D A8 C0 3C 6A 90  : AF
00E8   E9 C3 36 34 29 2B 52 44  : 00
00F0   15 E3 99 05 7F 7F A0 A6  : DA
00F8   19 B5 21 FE C0 BB EC 79  : CD
-----------------------------------
SUM:   94 5D C0 FA 20 08 C6 F3  6B54

0100   2C 7F 1D 7C DF 45 1C 2D  : B1
0108   D8 A2 F6 FF EA 6F 36 09  : 07
0110   55 7D 70 68 19 E6 86 F4  : 23
0118   80 FE 75 9D F3 26 68 47  : 58
0120   BD EE EF 6B D0 E5 20 0C  : E6
0128   61 C1 99 1A 10 2E 9B 88  : 36
0130   5D 94 17 9B AF F6 40 A0  : 28
0138   01 B4 89 B8 7C 97 FF 5C  : 64
0140   54 59 22 9F D0 D0 7B F8  : 81
0148   6E 6F 5F EA F3 93 9D 2F  : 78
0150   55 FC 00 FC F6 7A 32 07  : F6
0158   6F 67 F2 10 DE 0F 9A B0  : 0F
0160   15 C2 33 FC D5 C3 C1 26  : 85
0168   70 25 C0 0E FE 03 7D 95  : 76
0170   23 F4 9A E0 7E 5C C2 7B  : A8
0178   80 B0 FB 83 BC 4B 75 80  : AA
-----------------------------------
SUM:   03 49 1B 5A 84 B9 93 95  FD22

0180   D1 3D 49 A9 48 1E 97 40  : 3D
0188   E0 2E 3B 68 7E F8 82 76  : 1F
0190   9B CA D7 B5 C8 4F B0 39  : F1
0198   02 11 DB D7 67 2F 1E 9E  : 17
01A0   CD B9 CB DD D8 16 C3 57  : 36
01A8   6D 89 60 D7 33 EE 7A 46  : 0E
01B0   05 FE 5C 02 84 34 01 7E  : 98
01B8   BA 40 BD A2 E0 54 33 E2  : A2
01C0   51 4C 0B D3 18 07 0B 00  : A5
01C8   F2 98 07 65 80 7E A3 EF  : 86
01D0   70 D2 51 5B 42 F1 0C 51  : 7E
01D8   E7 32 3E CC 46 6B 01 C8  : 9D
01E0   52 3C 74 19 D5 70 CF FC  : 2B
01E8   E8 CD F7 0C FF 49 00 CB  : CB
01F0   2E 19 98 0C B8 33 DC 3C  : EE
01F8   C3 20 97 4D 9D B9 1B 89  : C1
-----------------------------------
SUM:   0C F0 B5 D2 AD A6 D9 1E  A446

0200   38 61 91 8D A0 A0 18 B3  : C2
0208   96 A1 36 1E C2 8B FB 68  : 3B
0210   05 ED 58 81 77 35 6C 89  : 6C
0218   51 AB EB 25 0B B9 A7 7C  : F3
0220   75 46 9D D2 C1 71 4C 6A  : 12
0228   64 1C 85 4F 85 29 F6 AC  : A4
0230   70 72 AC 47 1D C3 60 5E  : 73
0238   5B AD 8C 07 90 D8 C7 24  : EE
0240   83 3C 36 BB 40 29 B5 92  : 60
0248   43 45 29 EA C0 B9 A0 B0  : 64
0250   A4 1A 2E 61 6D E5 5D B8  : B4
0258   D6 89 6D 5A 72 78 F4 00  : 04
0260   4F 72 5D 8C 09 F6 BB 2D  : 91
0268   76 BD 1C 36 89 C0 AD EA  : 65
0270   2A FF 04 96 44 A3 89 76  : A9
0278   74 8A C2 E4 87 36 D9 78  : B2
-----------------------------------
SUM:   6B F7 9D 5C 13 1C FF B7  FC51

0280   A8 DA 04 EF 30 2B B1 1C  : 9D
0288   59 F6 AD F0 6A F0 10 5D  : B3
0290   DD 3E 6A BF 75 73 43 23  : 92
0298   56 46 25 1B C2 B7 11 BC  : 22
02A0   3B 40 7E E2 D8 A1 AD 85  : 86
02A8   02 1A 8E 75 2A D0 D7 7C  : 6C
02B0   53 79 71 58 AE AE FA 27  : 12
02B8   0C 89 84 76 1C 5A 0E 2A  : 3D
02C0   20 B8 BF 97 58 B7 0A 8F  : D6
02C8   72 6C FE 4C C0 4D 2F 96  : FA
02D0   7F C1 31 60 13 E8 4D 38  : 51
02D8   98 01 FA 57 05 5C 5F 93  : 3D
02E0   3C 1B 4A FC 83 59 FD 40  : B6
02E8   41 8B 10 41 B5 93 FA 20  : 7F
02F0   FF 13 FA 20 ED C1 20 D0  : CA
02F8   B8 26 19 22 07 2A 11 03  : 5E
-----------------------------------
SUM:   AD 75 96 F7 F9 DD AE CD  125C

0300   6B 28 20 42 66 40 9A F0  : 25
0308   40 A7 E8 81 A1 05 17 4D  : 5A
0310   78 DB 6B 47 81 67 BF 21  : CD
0318   20 24 AD A1 0C A1 76 3E  : F3
0320   42 13 DE 37 CA 10 98 C1  : 9D
0328   C9 B1 04 CB 86 EC 4C 86  : 8D
0330   F8 39 06 D4 09 06 08 44  : 66
0338   1B 34 4C 83 65 0C 94 41  : 64
0340   AC 84 83 D6 12 72 0C A8  : C1
0348   6F B9 C8 10 36 04 0E FB  : 43
0350   81 A4 09 A5 08 16 62 C4  : 17
0358   0D 8F 04 0C D0 C4 34 B2  : 26
0360   06 54 36 50 E8 A5 4C 0F  : C8
0368   D2 5C A8 1D 3E 7B A8 1F  : 73
0370   A6 F9 55 2D 3D 15 4F E7  : A9
0378   A2 72 8E 84 36 60 E4 83  : 23
-----------------------------------
SUM:   2A 8A 6D B9 0B 40 3D 19  4EB9

0380   5C 72 0C 07 64 19 1E 10  : 8C
0388   61 FE 20 EA 3E 20 C2 1C  : A5
0390   83 42 1B 4A 22 07 8D 04  : E4
0398   0D D4 44 40 BC BC 10 36  : 23
03A0   F6 08 81 C0 F4 40 DD 59  : A9
03A8   44 35 13 73 A0 B7 34 A9  : 33
03B0   5A 86 FC A9 5B 91 C6 A0  : D7
03B8   7C 9F 35 09 B6 CC AA 71  : F6
03C0   FE 82 AA 6D 86 41 36 83  : 17
03C8   C0 9A DE 64 0C FA 88 18  : 42
03D0   74 1F E9 C0 BA 7D A0 FF  : 12
03D8   7A 83 DB 04 7E 7A 0F 5A  : 3D
03E0   4B 8E C9 60 46 6F 71 DF  : 07
03E8   B6 A3 BB 47 D1 07 73 17  : BD
03F0   23 B5 B1 87 7A 1E 47 7A  : 69
03F8   20 49 68 C1 84 3E 73 0C  : D3
-----------------------------------
SUM:   4D D5 39 E4 04 54 09 E9  FE66

0400   68 75 85 0B 3B 81 E7 00  : 10
0408   6A DC 01 8F 80 33 C6 00  : 4F
0410   E7 70 06 46 80 FD FA 03  : 1D
0418   93 E8 0E 77 40 68 76 80  : 9E
0420   D5 AD 01 AA D0 1D 9D 01  : B8
0428   C5 5A 03 EF 0D 87 D6 5F  : DA
0430   CF C0 0D 80 E5 1F 77 6C  : 03
0438   07 86 10 22 D8 0F 10 47  : FD
0440   E4 A8 1E CB 15 2A 7B 81  : B0
0448   FC F8 B9 AE 07 5D B8 5E  : D5
0450   8A D7 5E 7F BF 5E 00 ED  : 48
0458   37 DF AC 01 D9 C0 FB 34  : 8B
0460   03 FE F7 45 D6 28 44 A9  : 28
0468   21 E7 EC 0E 89 17 35 00  : D7
0470   ED 18 8C FD 7B 03 B5 CF  : 90
0478   98 AD 78 62 F6 94 87 72  : A2
-----------------------------------
SUM:   06 F6 83 3D 99 66 FA 80  2BCC

0480   8C 31 17 A6 28 B2 06 F9  : 53
0488   40 63 E2 7B 1D 09 A5 4C  : 17
0490   15 A9 1E 0A 15 A4 BB AF  : 09
0498   B8 7D C4 E4 F4 60 72 4D  : F0
04A0   2B 4A 38 0D C0 9E 85 FA  : 97
04A8   6B 05 F3 52 A2 27 B0 9D  : CB
04B0   A7 FB 59 7C 90 FC FF A0  : A2
04B8   B7 FB B7 6C BF 96 70 22  : BC
04C0   C8 C9 25 9B 6B 2C 9F BE  : 45
04C8   48 50 26 F3 DE 5A 71 FD  : 57
04D0   77 6D 23 45 6F 04 5E C9  : E6
04D8   AE DA 3B AA 65 FF 73 0F  : 53
04E0   E2 0C 9E 62 5F 02 28 17  : 8E
04E8   36 F2 0A 0B CA 81 75 3C  : 39
04F0   F0 CB 65 D3 68 4D 05 A3  : 50
04F8   63 A6 BA D2 78 14 29 56  : A0
-----------------------------------
SUM:   2D CE 86 E5 25 83 28 79  BF8C

0500   5E 4E 80 C7 E6 7B 9E 04  : F6
0508   5A CF B6 81 CD 92 71 F6  : 26
0510   EF E6 E6 FA B1 DD CB EF  : FD
0518   7B 71 76 D4 4C C3 E3 B5  : DD
0520   EB 81 30 ED 73 7F 0B F2  : 78
0528   55 97 F6 98 0F 9D 7B 5D  : FE
0530   CE ED 68 8D 49 C2 3F D6  : D0
0538   7F 73 36 6E F9 A3 BF EF  : E0
0540   34 76 02 5A 3B 8E F6 85  : 4A
0548   EB F8 65 EE 51 D7 54 77  : 29
0550   6A E7 B3 72 AA DB 7C 72  : E9
0558   FF 8F 18 FE E1 AD 55 74  : FB
0560   C8 1E DF 6F CF 4D 2B 46  : C1
0568   1D 02 B7 AA C7 85 BB 5B  : E2
0570   90 47 95 C0 B4 2C 01 6F  : 7C
0578   E0 0A F5 C0 16 59 A0 E7  : 95
-----------------------------------
SUM:   8C 41 A8 E7 EB 72 E3 8B  911F

0580   C0 E3 40 6A B4 07 76 0A  : 88
0588   58 7B 48 86 B7 56 00 A0  : 4E
0590   9C 81 7E CC 81 4B 1A 81  : CE
0598   63 D3 20 55 FC 81 46 E4  : 52
05A0   0B 62 C0 0E 0B F8 05 7A  : BD
05A8   70 0A F3 E0 15 E5 A8 17  : 06
05B0   4B FA 81 5E 08 0A 84 C0  : 7A
05B8   B9 AD 12 D8 9F CC 18 15  : E8
05C0   4E 63 B4 46 05 B8 30 2A  : C2
05C8   A3 0B E8 8F 45 3A 23 D0  : 97
05D0   2E 88 F4 73 A2 3D 02 81  : 7F
05D8   18 02 81 18 02 B7 88 58  : 4C
05E0   03 FC 09 5E A4 D2 92 E6  : 54
05E8   92 DE 09 1B D8 24 2B 42  : FD
05F0   3E B2 03 F9 C0 21 63 02  : 32
05F8   B4 33 D8 4E D3 0A B7 BC  : 5D
-----------------------------------
SUM:   54 7C 6A 55 AC E3 D3 2E  EFA4

0600   9C 09 F6 A2 59 F5 F3 AD  : 2B
0608   E7 B1 14 C7 A0 1D E7 42  : 59
0610   33 95 C0 05 00 32 11 20  : F0
0618   50 2B 4A 48 13 7A 64 9F  : 9D
0620   33 7A 2B 26 63 19 81 D1  : CC
0628   6A FB EC C4 44 AB E9 2A  : 17
0630   EC 05 4C 26 6E 31 5E 91  : F1
0638   38 04 8F DD F6 24 FF 01  : C2
0640   6C 4C FB 02 F6 02 C5 C9  : 3B
0648   B2 16 89 77 D4 83 5A 24  : 9D
0650   EB 01 42 78 09 1B 63 AF  : DC
0658   C8 23 62 88 DB 6D 9B 7F  : 37
0660   59 23 DB 4D FB AD 7C 21  : E9
0668   1A B6 9B E8 04 8F 9E 48  : CC
0670   FF 6E ED 04 8D 49 22 93  : E9
0678   BA 38 E8 49 5A 8A C7 B9  : 87
-----------------------------------
SUM:   C4 FD 79 9E AB F3 36 0B  1FCD

0680   27 80 46 FC 56 3F A9 22  : 49
0688   C4 48 84 C5 C3 12 34 CB  : 29
0690   1D 59 2B AD CE 76 4F 06  : E7
0698   D8 39 E1 92 B6 E4 AB DA  : A3
06A0   A4 E9 12 B8 24 A8 1D 04  : 44
06A8   EB E4 44 CF 7E 18 6E 05  : EB
06B0   BB A1 DD 18 61 C9 86 17  : 18
06B8   76 FA 1D 42 8E 34 37 BE  : 86
06C0   E0 3A D5 9A AF B0 02 E8  : D2
06C8   23 80 3F 1D 30 07 53 37  : C0
06D0   2F 36 6D 24 05 B0 22 05  : D2
06D8   90 01 0B DC 0B A2 D0 23  : 18
06E0   60 7E 05 A0 D7 8D 21 03  : 0B
06E8   BD 90 17 0B 20 5F ED 41  : 1C
06F0   1E B7 F0 15 43 BB F5 0E  : DB
06F8   E3 AA 0B 4A 28 5F 0C 50  : C5
-----------------------------------
SUM:   80 22 C9 A2 7F 77 75 94  0CE1

0700   58 17 F0 2A 8D 47 C6 0B  : 2E
0708   A8 14 3D 51 F8 F6 C1 DD  : D6
0710   6B 84 4E 0D 78 05 CB 70  : 02
0718   8E FF 0F F9 C9 38 3B 9A  : 6B
0720   CE 05 AE 70 8E 94 E0 BF  : B2
0728   A5 D3 D1 37 17 D4 70 5D  : 38
0730   57 0B D0 40 14 52 00 58  : 30
0738   27 C5 C0 8E A2 00 B7 B0  : 43
0740   11 BB 72 0B 39 50 05 C8  : 9F
0748   A6 FA 48 16 AB 24 FF B8  : 84
0750   1B A0 23 21 D3 E9 C8 5B  : DE
0758   8C 8B F3 7C 97 8A 1F 20  : E6
0760   03 DD CC 4E CE 3E 7D 8C  : 0F
0768   8D FF 94 1F A1 60 3C 60  : DC
0770   63 8E 42 7E 45 09 64 1C  : 7F
0778   84 F8 1C 84 E4 12 15 FC  : 23
-----------------------------------
SUM:   BF 98 27 23 07 D4 B1 15  91CA

0780   E4 27 FC A4 A9 63 91 7C  : C4
0788   3C E4 5D 8E 72 2F F2 8C  : 2A
0790   F6 40 3F 64 03 B1 20 1C  : C9
0798   12 01 CA 88 06 C8 80 6A  : 1D
07A0   44 02 39 0F 51 9B 48 07  : C9
07A8   F6 40 38 31 00 AA 85 14  : E2
07B0   1E 6B C5 40 80 47 D1 6D  : 93
07B8   84 D8 BB B0 CB 20 11 08  : CB
07C0   03 E9 40 10 C8 03 66 50  : BD
07C8   07 01 00 6C 68 F1 14 61  : 42
07D0   07 7E 00 A0 00 D0 00 D6  : CB
07D8   00 EC 00 AA 00 0F 4C 4C  : 3D
07E0   5F A5 37 FD CD AE C0 24  : 97
07E8   CE 2E C2 2C 02 F8 09 3A  : 27
07F0   E0 3F F8 8C 00 3F 00 3D  : 1F
07F8   9B 3B 58 03 83 98 07 EB  : 3E
-----------------------------------
SUM:   BD 72 DC CC 42 07 68 77  F35F

0800   F3 04 1F DA 0F FE 0D 8F  : 99
0808   20 D9 09 24 59 7F 2C 67  : 91
```

```
0810  70 61 00 5A 2F C4 00 76  : 94
0818  8F 7D CD 29 F4 56 D0 6F  : 8B
0820  A4 CF E9 06 B7 21 CF B7  : C0
0828  92 0D 11 B8 9C 38 FD 28  : 61
0830  BC C3 87 F4 B3 ED 37 5E  : 2F
0838  08 FC 63 F2 DF F2 12 F0  : 2C
0840  3D DD 1A 60 20 9B 09 23  : 7B
0848  FE 6F FF A4 3F 20 79 12  : FA
0850  1C 37 E0 DF 38 01 F9 33  : 77
0858  78 4F 1A 52 1A 4E 1C F0  : A7
0860  01 7E 56 F2 84 0F FD 87  : DE
0868  E2 46 FD 6A B6 C5 53 9E  : FB
0870  C3 F2 5B E0 45 82 4A 4F  : 50
0878  08 7F 43 44 05 A4 40 92  : 89
------------------------------------
SUM:  89 5D DD DA A5 D3 8F 66  6038

0880  66 42 92 27 CA D1 48 93  : D7
0888  32 87 42 8E 37 9B AA 03  : 08
0890  FF C5 A4 95 6A 7B AD 15  : A4
0898  21 D4 F7 52 1F 85 62 A1  : E5
08A0  DD E4 06 BD 0F E0 EB 0E  : 6C
08A8  1B EA 6B B2 B7 E1 5A F6  : 0A
08B0  97 F8 4F F8 0F F4 87 62  : C2
08B8  4D 80 0A F0 02 B2 C9 60  : A4
08C0  03 16 3B 86 DF A6 EB 60  : AA
08C8  87 D2 BD 40 1B 74 E5 95  : 5F
08D0  5B 92 6E CA 7C D7 FB B2  : 25
08D8  FF 8F 9B 50 8F 75 72 EA  : D9
08E0  6F D4 A3 2E E4 7E 62 F0  : C8
08E8  0D 4F 23 98 5A EB E5 0D  : 4E
08F0  1F 67 6C F9 B2 8D 45 DE  : 4D
08F8  E9 91 E3 7B BA B0 74 63  : 19
------------------------------------
SUM:  FC CC 4F 0D 10 DF D3 E1  8D24

0900  46 D0 EE EC A1 43 33 4A  : 51
0908  D7 58 33 65 8A 49 B7 48  : 99
0910  8F 5A 17 DB 05 4A A7 54  : 25
0918  67 50 F5 49 0D BB 9B 92  : EA
0920  B6 C3 36 97 FE 3D F1 60  : D2
0928  06 3E 1B 5C 2C 62 2D 9B  : 11
0930  DB 77 EE 17 39 15 67 0E  : 1A
0938  B5 27 DA 4F 36 C2 DE DD  : B8
0940  2F AD 09 A3 17 A2 37 91  : 09
0948  46 D3 67 6C 90 B8 3B AA  : 19
0950  B5 BD B9 EB C0 3E DA 5C  : 4A
0958  70 6B 57 0F 53 5F 54 FE  : 45
0960  C9 7A 1C 8C 71 BE E3 31  : 2E
0968  C5 6B 69 CF D8 13 BB 13  : 21
0970  F6 08 E4 D0 DC 99 C6 2A  : 17
0978  8A 7A 79 7A 3D 00 FD 4B  : 7C
------------------------------------
SUM:  07 80 A8 7C F2 68 90 AC  8054

0980  F5 C2 C7 C0 8F C2 9E E7  : 14
0988  8E 1B 79 EF EB 20 A3 C5  : 84
0990  FB 77 03 90 6F 4A BF 54  : D1
0998  3F A6 3E E8 23 FB 2D 53  : A9
09A0  02 75 48 7A D7 BE 9F FE  : 6B
09A8  28 3C D7 1E 3A 3D D0 EF  : 8F
09B0  18 9F 67 0D E8 8F C0 EF  : 51
09B8  2A AF 2C 0F 04 FC 91 5C  : 01
09C0  1F E3 FF E2 47 B0 57 FC  : 2D
09C8  4D FE 1D 3A 09 D4 CF C9  : 17
09D0  2D FC 60 FA 07 DF BC F5  : 1A
09D8  AD EB 31 E5 A6 B1 B2 1D  : D4
09E0  05 F5 8C F5 2F E4 8B 40  : 59
09E8  37 EB E9 B7 5F 2C 77 61  : 25
09F0  0F 42 74 3D FB F1 03 EA  : DB
09F8  93 EC 05 0B 0D BD 91 FA  : E4
------------------------------------
SUM:  4D CF CE CA 9C 7F 17 E7  9815

0A00  17 CD DF D8 9C 7A 4A B3  : AE
0A08  15 01 88 8F B0 ED F9 DD  : A0
0A10  7D E4 BD BE 99 FD A1 31  : 44
0A18  97 3D A3 FD A0 31 DA AF  : CE
0A20  6B 5C 78 4F C4 48 FD B4  : 4B
0A28  3D 21 75 19 D5 96 36 D7  : 64
0A30  F3 0F E0 1D B8 7F DC 58  : 6A
0A38  51 99 10 A4 07 DC 2E FF  : AE
0A40  48 FE DB 89 49 EA 3A 7B  : 92
0A48  A2 72 90 34 66 77 5A 9E  : AD
0A50  AD 9B A6 BE A0 2C 14 BF  : 4B
0A58  A8 3E 9C 25 92 34 50 8A  : 47
0A60  50 F2 9C F2 35 D3 63 BC  : F7
0A68  56 94 90 DD 6F 2A 74 BA  : 1E
0A70  48 3A EF 95 60 92 24 4D  : 69
0A78  E3 F1 02 BB 21 83 A7 67  : 43
------------------------------------
SUM:  3C 0E 6E 0A E3 A1 95 DE  E2FC

0A80  56 38 E9 A2 7F 0B FB 0B  : A9
0A88  F1 CE 3C 6A 5F 81 B3 79  : 71
0A90  B3 21 E8 FC CF 07 F4 0C  : 8E
0A98  DC E9 B6 A5 1C BC BD D9  : 8E
0AA0  CD E5 C6 2D C2 B5 3A AB  : 01
0AA8  26 9B FC 48 9C 87 DB 1C  : 1F
0AB0  27 20 E6 47 9C C1 9E 4D  : BC
0AB8  F8 9B E0 09 A4 FD 85 33  : D5
0AC0  94 3C F9 1F CE A0 FB 58  : A9
0AC8  6E 87 0F A7 3B BA 5D FD  : FA
0AD0  B1 DE 94 6E 49 2F 4B 43  : 97
0AD8  49 51 FB B0 D4 44 6D DC  : A6
0AE0  BE 7D BC BA 6D A6 3E AD  : AF
0AE8  DB 49 1A 0B 4D B5 EC F8  : 2F
0AF0  2F 52 2A E3 51 FE CC 54  : FD
0AF8  99 56 B7 8B A1 C9 10 A4  : 4F
------------------------------------
SUM:  45 AB 99 89 39 38 AD C1  DD98

0B00  9E 47 D7 18 78 31 D4 81  : D2
0B08  C9 16 AF DB 68 6C B3 BB  : AB
0B10  7F 10 BB 0D 7F 21 5F 33  : 89
0B18  37 C0 17 C0 19 00 1A 00  : 01
0B20  12 14 00 4D D6 98 35 5B  : 71
0B28  A2 40 39 A4 D8 BD EF 6A  : AD
0B30  BA DA 2D 79 09 B6 61 F8  : 52
0B38  3B E4 AC 00 85 F9 9A E0  : C3
0B40  F5 75 75 F4 DF 42 A8 A9  : 45
0B48  FC EF C5 1F F2 B3 D6 A2  : EC
0B50  AF 2D 28 66 B0 1C 2B 4A  : AB
0B58  98 FD A6 26 8C 62 00 19  : 68
0B60  6C FA A3 6E 28 A9 26 22  : 90
0B68  5A F5 AF 10 22 A2 1E 31  : 21
0B70  38 3C 81 28 78 49 F6 00  : D4
0B78  4A 64 85 9A B9 BA A4 A5  : 89
------------------------------------
SUM:  46 5C CA 09 3C 83 A6 B2  E9B4

0B80  67 F2 6D 90 C0 83 C0 D8  : 31
0B88  FC 54 5F F0 CE EF F7 34  : 87
0B90  FB 81 F7 F8 15 21 FE F3  : 92
0B98  E9 FF C4 75 2F 02 80 5B  : 2D
0BA0  3C 1B 3F B6 73 D7 FB EB  : 7C
0BA8  D1 96 70 81 34 6F 1C 4E  : 65
0BB0  87 00 6C 79 12 6C FF 99  : 82
0BB8  72 22 D4 15 08 00 00 00  : 85
0BC0  00 00 00 00 00 00 00 00  : 00
0BC8  00 00 00 00 00 00 00 00  : 00
0BD0  00 00 00 00 00 00 00 00  : 00
0BD8  00 00 00 00 00 00 00 00  : 00
0BE0  00 00 00 00 00 00 00 00  : 00
0BE8  00 00 00 00 00 00 00 00  : 00
0BF0  00 00 00 00 00 00 00 00  : 00
0BF8  00 00 00 00 00 00 00 00  : 00
------------------------------------
SUM:  4D 99 76 B2 93 47 4B 2C  F1B0
```

## リスト2

```
  1:          .include    doscall.mac
  2:          .include    iocscall.mac
  3:          .include    magic.mac
  4:
  5:          .text
  6:          .even
  7:
  8:          move.w  #8,d1
  9:          IOCS    _CRTMOD
 10:          IOCS    _B_CUROFF
 11:          IOCS    _OS_CUROF
 12:
 13:          lea.l   initdata,a0
 14:          MAGIC   __AUTO
 15:
 16:          lea.l   k,a3
 17: start:
 18:          clr.l   (a3)
 19:          clr.l   4(a3)
 20:          clr.l   8(a3)
 21:          move.w  #3,12(a3)
 22:          move.w  #3,14(a3)
 23:          bsr     MAZEGET
 24:
 25:          clr.w   d1
 26:          move.w  #16,d2
 27:          IOCS    _B_LOCATE
 28:          lea.l   TITLEPTR,a1
 29:          IOCS    _B_PRINT
 30:
 31: start2:
 32:          lea.l   para,a0
 33:          move.w  40(a0),d0
 34:          subq.w  #1,d0
 35:          move.w  d0,40(a0)
 36:          MAGIC   __AUTO
 37:          lea.l   WAKU,a0
 38:          MAGIC   __AUTO
 39:          lea.l   pers,a0
 40:          MAGIC   __AUTO
 41:          lea.l   disp,a0
 42:          MAGIC   __AUTO
 43:          bsr     BREAK
 44:          bsr     ACHECK
 45:          bne     start2
 46:
 47: start3:  lea.l   para,a0
 48:          clr.w   40(a0)
 49:          bsr     cls
 50:          IOCS    _TIMEGET
 51:          move.l  d0,d1
 52:          IOCS    _TIMEBIN
 53:          lea.l   TIME,a0
 54:          move.l  d0,(a0)
 55:
 56: loop:
 57:          bsr     JOY
 58:          btst    #6,d0
 59:          beq     gameover
 60:
 61:          bsr     rotatedo
 62:          bsr     slidedo
 63:          bsr     rotate
 64:          bsr     slide
 65:          bsr     haichi
 66:          bsr     GOALHYOUJI
 67:
 68:          lea.l   disp,a0
 69:          MAGIC   __AUTO
 70:          cmpi.w  #36,12(a3)
 71:          bne     loop2
 72:          cmpi.w  #36,14(a3)
 73:          beq     clear
 74: loop2:
 75:          bsr     PAUSE
 76:          bsr     BREAK
 77:          bsr     TIMEPRINT
 78:          bra     loop
 79: clear:
 80:          bsr     rotate
 81:          bsr     slide
 82:          bsr     haichi
 83:          bsr     GOALHYOUJI
 84:          lea.l   disp,a0
 85:          MAGIC   __AUTO
 86:
 87:          tst.w   (a3)
 88:          bne     clear
 89:
 90:          clr.w   d1
 91:          move.w  #19,d2
 92:          IOCS    _B_LOCATE
 93:          lea.l   CLEARPTR,a1
 94:          IOCS    _B_PRINT
 95: gameover:
 96:          bsr     game_over
 97:          bra     start
 98:
 99: JOY:     clr.w   d1
100:          IOCS    _JOYGET
101:          rts
102:
103: GOALHYOUJI:
104:          lea.l   para,a0
105:          movea.l a0,a1
106:          move.w  12(a3),d1
107:          sub.w   #36,d1
108:          muls    #20,d1
109:          move.w  d1,22(a1)
110:
111:          move.w  14(a3),d1
112:          neg.w   d1
113:          add.w   #36,d1
114:          muls    #20,d1
115:          move.w  d1,34(a1)
116:          MAGIC   __AUTO
117:          clr.w   22(a1)
118:          clr.w   34(a1)
119:          lea.l   GOAL,a0
120:          MAGIC   __AUTO
121:          lea.l   pers,a0
122:          MAGIC   __AUTO
123:          rts
124: rotatedo:
125:          btst    #2,d0
126:          beq     rotatedo2
127:          btst    #3,d0
128:          beq     rotatedo5
129:          rts
130: rotatedo2:
131:          tst.w   6(a3)
132:          bmi     rotatedo_ret
133:
134:          move.w  4(a3),d2
135:          move.w  d2,2(a3)
136:          subq.w  #1,4(a3)
137:
138:          bpl     rotatedo8
139:
140:          addq.w  #4,2(a3)
141:          addq.w  #4,4(a3)
142:          bra     rotatedo8
143: rotatedo5:
144:          tst.w   6(a3)
145:          beq     rotatedo6
146:          bpl     rotatedo_ret
147: rotatedo6:
148:          move.w  4(a3),d2
149:          move.w  d2,2(a3)
150:          addq.w  #1,4(a3)
```



▶アイビット電子の広告がまたつまらなくなったので，いまはOAシステムプラザの広告にあるカットを楽しみにしています。　石田　伯仁(18)神奈川県

```
151:         cmpi.w  #5,4(a3)
152:
153:         bne     rotatedo8
154:         subq.w  #4,2(a3)
155:         subq.w  #4,4(a3)
156: rotatedo8:
157:         move.w  2(a3),d2
158:         muls    #90,d2
159:         move.w  d2,8(a3)
160:         move.w  4(a3),d2
161:         muls    #90,d2
162:         move.w  d2,10(a3)
163:         tst.w   6(a3)
164:         beq     rotatedo13
165:         bpl     rotatedo12
166:
167:         addi.w  #90,6(a3)
168:         rts
169: rotatedo12:
170:         subi.w  #90,6(a3)
171:         rts
172: rotatedo13:
173:         sub.w   8(a3),d2
174:         move.w  d2,6(a3)
175: rotatedo_ret:
176:         rts
177:
178: slidedo:
179:         tst.w   (a3)
180:         bne     slidedo_ret
181:         tst.w   6(a3)
182:         bne     slidedo_ret
183:         move.w  4(a3),d2
184:         lea.l   h,a2
185:         btst    #0,d0
186:         beq     slidef
187:         btst    #1,d0
188:         beq     slideb
189:         rts
190: slideb:
191:         tst.w   d2
192:         beq     slideb0
193:         subq.w  #1,d2
194:         beq     slideb1
195:         subq.w  #1,d2
196:         beq     slideb2
197:         subq.w  #1,d2
198:         beq     slideb3
199:         subq.w  #1,d2
200:         beq     slideb0
201:         subq.w  #1,d2
202:         beq     slideb1
203:         rts
204: slidef:
205:         tst.w   d2
206:         beq     slidef0
207:         subq.w  #1,d2
208:         beq     slidef1
209:         subq.w  #1,d2
210:         beq     slidef2
211:         subq.w  #1,d2
212:         beq     slidef3
213:         subq.w  #1,d2
214:         beq     slidef0
215:         subq.w  #1,d2
216:         beq     slidef1
217:         rts
218: slidef0:
219:         move.w  48(a2),d2
220:         btst    #0,d2
221:         bne     slidedo_ret
222:
223:         subq.w  #1,14(a3)
224:         bsr     MAZEGET
225:         move.w  #20,(a3)
226:         rts
227: slidef1:
228:         move.w  48(a2),d2
229:         btst    #1,d2
230:         bne     slidedo_ret
231:
232:         addq.w  #1,12(a3)
233:         bsr     MAZEGET
234:         move.w  #20,(a3)
235:         rts
236: slidef2:
237:         move.w  48(a2),d2
238:         btst    #2,d2
239:         bne     slidedo_ret
240:
241:         addq.w  #1,14(a3)
242:         bsr     MAZEGET
243:         move.w  #20,(a3)
244:         rts
245: slidef3:
246:         move.w  48(a2),d2
247:         btst    #3,d2
248:         bne     slidedo_ret
249:
250:         subq.w  #1,12(a3)
251:         bsr     MAZEGET
252:         move.w  #20,(a3)
253:         rts
254: slideb0:
255:         move.w  48(a2),d2
256:         btst    #2.d2
257:         bne     slidedo_ret
258:
259:         addq.w  #1,14(a3)
260:         bsr     MAZEGET
261:         move.w  #-20,(a3)
262:         rts
263: slideb1:
264:         move.w  48(a2),d2
265:         btst    #3,d2
266:         bne     slidedo_ret
267:
268:         subq.w  #1,12(a3)
269:         bsr     MAZEGET
270:         move.w  #-20,(a3)
271:         rts
272: slideb2:
273:         move.w  48(a2),d2
274:         btst    #0,d2
275:         bne     slidedo_ret
276:
277:         subq.w  #1 4(a3)
278:         bsr     MAZEGET
279:         move.w  #-20,(a3)
280:         rts
281: slideb3:
282:         move.w  48(a2),d2
283:         btst    #1,d2
284:         bne     slidedo_ret
285:
286:         addq.w  #1,12(a3)
287:         bsr     MAZEGET
288:         move.w  #-20,(a3)
289:         rts
290: MAZEGET:
291:         lea.l   MAZE,a6
292:         lea.l   h,a2
293:         moveq.l #0,d1
294:         moveq.l #0,d2
295:         move.w  12(a3),d1
296:         move.w  14(a3),d2
297:         subq.w  #3,d1
298:         subq.w  #3,d2
299:         muls    #2,d1
300:         muls    #80,d2
301:
302:         add.w   d1,d2
303:         add.l   d2,a6
304:
305:         move.l  (a6)+,(a2)+
306:         move.l  (a6)+,(a2)+
307:         move.l  (a6)+,(a2)+
308:         move.w  (a6),(a2)+
309:         adda.w  #68,a6
310:
311:         move.l  (a6)+,(a2)+
312:         move.l  (a6)+,(a2)+
313:         move.l  (a6)+,(a2)+
314:         move.w  (a6),(a2)+
315:         adda.w  #68,a6
316:
317:         move.l  (a6)+,(a2)+
318:         move.l  (a6)+,(a2)+
319:         move.l  (a6)+,(a2)+
320:         move.w  (a6),(a2)+
321:         adda.w  #68,a6
322:
323:         move.l  (a6)+,(a2)+
324:         move.l  (a6)+,(a2)+
325:         move.l  (a6)+,(a2)+
326:         move.w  (a6),(a2)+
327:         adda.w  #68,a6
328:
329:         move.l  (a6)+,(a2)+
330:         move.l  (a6)+,(a2)+
331:         move.l  (a6)+,(a2)+
332:         move.w  (a6),(a2)+
333:         adda.w  #68,a6
334:
335:         move.l  (a6)+,(a2)+
336:         move.l  (a6)+,(a2)+
337:         move.l  (a6)+,(a2)+
338:         move.w  (a6),(a2)+
339:         adda.w  #68,a6
340:
341:         move.l  (a6)+,(a2)+
342:         move.l  (a6)+,(a2)+
343:         move.l  (a6)+,(a2)+
344:         move.w  (a6),(a2)
345: slidedo_ret:
346:         rts
347: slide:
348:         move.w  (a3),d2
349:         tst.w   d2
350:         beq     slide_ret
351:         bpl     slide2
352:         addq.w  #2,(a3)
353:         add.w   #12,d2
354:         bra     slide3
355: slide2:
356:         subq.w  #2,(a3)
357:         addq.w  #8,d2
358: slide3:
359:         lea.l   para,a0
360:         move.w  d2,16(a0)
361:         MAGIC   __AUTO
362: slide_ret:
363:         rts
364:
365: rotate:
366:         move.w  6(a3),d3
367:         tst.w   d3
368:         beq     rotate_ret
369:         bpl     rotate2
370:         addq.w  #5,d3
371:         bra     rotate3
372: rotate2:
373:         subq.w  #5,d3
374: rotate3:
375:         move.w  d3,6(a3)
376:         move.w  10(a3),d2
377:         sub.w   d3,d2
378:         lea.l   para,a0
379:         move.w  d2,40(a0)
380:         MAGIC   __AUTO
381: rotate_ret:
382:         rts
383:
384: haichi:
385:         lea.l   para,a1
386:         move.l  a1,a0
387:         lea.l   h.a2
388:
389:         move.w  16(a1),d1
390:         move.w  #10,16(a1)
391:         move.w  40(a1),d2
392:         clr.w   40(a1)
393:         MAGIC   __AUTO
394:         move.w  d1,16(a1)
395:         move.w  d2,40(a1)
396:         lea.l   TRIANGLE,a0
397:         MAGIC   __AUTO
398:         lea.l   pers,a0
399:         MAGIC   __AUTO
400:
401:         move.w  48(a2),d1
402:         move.w  d1,d2
403:         clr.w   22(a1)
404:         clr.w   34(a1)
405:         bsr     select_obj
406:         btst    #0,d2
407:         bne     H211
408:
409:         move.w  34(a2),d1
410:         move.w  d1,d2
411:         move.w  #-20,34(a1)
412:         bsr     select_obj
413:         btst    #3,d2
414:         bne     H113
415:
416:         move.w  32(a2),d1
417:         move.w  #20,22(a1)
418:         bsr     select_obj
419: H113:
420:         btst    #1,d2
421:         bne     H121
422:         move.w  36(a2),d1
423:         move.w  #-20,22(a1)
424:         bsr     select_obj
425: H121:
426:         btst    #0,d2
427:         bne     H211
428:         move.w  20(a2),d1
429:         move.w  d1,d2
430:         clr.w   22(a1)
431:         move.w  #-40,34(a1)
432:         bsr     select_obj
433:
434:         btst    #3,d2
435:         bne     H123
436:         move.w  18(a2),d1
437:         move.w  #20,22(a1)
438:         bsr     select obj
```

▶「SHIFT BREAK」の（哲）さん。「45度チョップ」とは，もしやのび太のママが得意とするあの技のことでしょうか？　国政　寛(20)大阪府

```
439: H123:
440:            btst     #1,d2
441:            bne      H131
442:            move.w   22(a2),d1
443:            move.w   #-20,22(a1)
444:            bsr      select_obj
445: H131:
446:            btst     #0,d2
447:            bne      H211
448:            move.w   6(a2),d1
449:            move.w   d1,d2
450:            clr.w    22(a1)
451:            move.w   #-60,34(a1)
452:            bsr      select_obj
453:
454:            btst     #3,d2
455:            bne      H133
456:            move.w   4(a2),d1
457:            move.w   #20,22(a1)
458:            bsr      select_obj
459: H133:
460:            btst     #1,d2
461:            bne      H211
462:            move.w   8(a2),d1
463:            move.w   #-20,22(a1)
464:            bsr      select_obj
465:
466: H211:
467:            move.w   48(a2),d1
468:            btst     #1,d1
469:            bne      H311
470:
471:            move.w   50(a2),d1
472:            move.w   d1,d2
473:            move.w   #-20,22(a1)
474:            clr.w    34(a1)
475:            bsr      select_obj
476:            btst     #0,d2
477:            bne      H213
478:
479:            move.w   36(a2),d1
480:            move.w   #-20,34(a1)
481:            bsr      select_obj
482: H213:
483:            btst     #2,d2
484:            bne      H221
485:            move.w   64(a2),d1
486:            move.w   #20,34(a1)
487:            bsr      select_obj
488: H221:
489:            btst     #1,d2
490:            bne      H311
491:            move.w   52(a2),d1
492:            move.w   d1,d2
493:            move.w   #-40,22(a1)
494:            clr.w    34(a1)
495:            bsr      select_obj
496:
497:            btst     #0,d2
498:            bne      H223
499:            move.w   38(a2),d1
500:            move.w   #-20,34(a1)
501:            bsr      select_obj
502: H223:
503:            btst     #2,d2
504:            bne      H231
505:            move.w   66(a2),d1
506:            move.w   #20,34(a1)
507:            bsr      select_obj
508: H231:
509:            btst     #1,d2
510:            bne      H311
511:            move.w   54(a2),d1
512:            move.w   d1,d2
513:            move.w   #-60,22(a1)
514:            clr.w    34(a1)
515:            bsr      select_obj
516:
517:            btst     #0,d2
518:            bne      H233
519:            move.w   40(a2),d1
520:            move.w   #-20,34(a1)
521:            bsr      select_obj
522: H233:
523:            btst     #2,d2
524:            bne      H311
525:            move.w   68(a2),d1
526:            move.w   #20,34(a1)
527:            bsr      select_obj
528: H311:
529:            move.w   48(a2),d1
530:            btst     #2,d1
531:            bne      H411
532:
533:            move.w   62(a2),d1
534:            move.w   d1,d2
535:            clr.w    22(a1)
536:            move.w   #20,34(a1)
537:            bsr      select_obj
538:            btst     #1,d2
539:            bne      H313
540:
541:            move.w   64(a2),d1
542:            move.w   #-20,22(a1)
543:            bsr      select_obj
544: H313:
545:            btst     #3,d2
546:            bne      H321
547:            move.w   60(a2),d1
548:            move.w   #20,22(a1)
549:            bsr      select_obj
550: H321:
551:            btst     #2,d2
552:            bne      H411
553:            move.w   76(a2),d1
554:            move.w   d1,d2
555:            clr.w    22(a1)
556:            move.w   #40,34(a1)
557:            bsr      select_obj
558:
559:            btst     #1,d2
560:            bne      H323
561:            move.w   78(a2),d1
562:            move.w   #-20,22(a1)
563:            bsr      select_obj
564: H323:
565:            btst     #3,d2
566:            bne      H331
567:            move.w   74(a2),d1
568:            move.w   #20,22(a1)
569:            bsr      select_obj
570: H331:
571:            btst     #2,d2
572:            bne      H411
573:            move.w   90(a2),d1
574:            move.w   d1,d2
575:            clr.w    22(a1)
576:            move.w   #60,34(a1)
577:            bsr      select_obj
578:
579:            btst     #1,d2
580:            bne      H333
581:            move.w   92(a2),d1
582:            move.w   #-20,22(a1)
583:            bsr      select_obj
584: H333:
585:            btst     #3,d2
586:            bne      H411
587:            move.w   88(a2),d1
588:            move.w   #20,22(a1)
589:            bsr      select_obj
590: H411:
591:            move.w   48(a2),d1
592:            btst     #3,d1
593:            bne      H_RET
594:
595:            move.w   46(a2),d1
596:            move.w   d1,d2
597:            move.w   #20,22(a1)
598:            clr.w    34(a1)
599:            bsr      select_obj
600:            btst     #2,d2
601:            bne      H413
602:
603:            move.w   60(a2),d1
604:            move.w   #20,34(a1)
605:            bsr      select_obj
606: H413:
607:            btst     #0,d2
608:            bne      H421
609:            move.w   32(a2),d1
610:            move.w   #-20,34(a1)
611:            bsr      select_obj
612: H421:
613:            btst     #3,d2
614:            bne      H_RET
615:            move.w   44(a2),d1
616:            move.w   d1,d2
617:            move.w   #40,22(a1)
618:            clr.w    34(a1)
619:            bsr      select_obj
620:
621:            btst     #2,d2
622:            bne      H423
623:            move.w   58(a2),d1
624:            move.w   #20,34(a1)
625:            bsr      select_obj
626: H423:
627:            btst     #0,d2
628:            bne      H431
629:            move.w   30(a2),d1
630:            move.w   #-20,34(a1)
631:            bsr      select_obj
632: H431:
633:            btst     #3,d2
634:            bne      H_RET
635:            move.w   42(a2),d1
636:            move.w   d1,d2
637:            move.w   #60,22(a1)
638:            clr.w    34(a1)
639:            bsr      select_obj
640:
641:            btst     #2,d2
642:            bne      H433
643:            move.w   56(a2),d1
644:            move.w   #20,34(a1)
645:            bsr      select_obj
646: H433:
647:            btst     #0,d2
648:            bne      H_RET
649:            move.w   28(a2),d1
650:            move.w   #-20,34(a1)
651:            bsr      select_obj
652: H_RET:
653:            clr.w    22(a1)
654:            clr.w    34(a1)
655:            rts
656:
657: select_obj:
658:            movea.l  a1,a0
659:            MAGIC    __AUTO
660:            tst.w    d1
661:            beq      s_obj0
662:            subq.w   #1,d1
663:            beq      s_obj1
664:            subq.w   #1,d1
665:            beq      s_obj2
666:            subq.w   #1,d1
667:            beq      s_obj3
668:            subq.w   #1,d1
669:            beq      s_obj4
670:            subq.w   #1,d1
671:            beq      s_obj5
672:            subq.w   #1,d1
673:            beq      s_obj6
674:            subq.w   #1,d1
675:            beq      s_obj7
676:            subq.w   #1,d1
677:            beq      s_obj8
678:            subq.w   #1,d1
679:            beq      s_obj9
680:            subq.w   #1,d1
681:            beq      s_objA
682:            subq.w   #1,d1
683:            beq      s_objB
684:            subq.w   #1,d1
685:            beq      s_objC
686:            subq.w   #1,d1
687:            beq      s_objD
688:            subq.w   #1,d1
689:            beq      s_objE
690:            subq.w   #1,d1
691:            beq      s_objF
692:
693: s_obj0:
694:            rts
695: s_obj1:
696:            lea.l    obj1,a0
697:            bra      s_rts
698: s_obj2:
699:            lea.l    obj2,a0
700:            bra      s_rts
701: s_obj3:
702:            lea.l    obj3,a0
703:            bra      s_rts
704: s_obj4:
705:            lea.l    obj4,a0
706:            bra      s_rts
707: s_obj5:
708:            lea.l    obj5,a0
709:            bra      s_rts
710: s_obj6:
711:            lea.l    obj6,a0
712:            bra      s_rts
713: s_obj7:
714:            lea.l    obj7,a0
715:            bra      s_rts
716: s_obj8:
717:            lea.l    obj8,a0
718:            bra      s_rts
719: s_obj9:
720:            lea.l    obj9,a0
721:            bra      s_rts
722: s_objA:
723:            lea.l    objA,a0
724:            bra      s_rts
725: s_objB:
726:            lea.l    objB,a0
```

▶今日，通帳記入したら残高があまりなかった。どうやらガス料金の引き落としができなかった気配……。う～ん，生活苦（自慢してどうする）。　升井　晋也(22)福岡県

```
727:            bra     s_rts
728: s_objC:
729:            lea.l   objC,a0
730:            bra     s_rts
731: s_objD:
732:            lea.l   objD,a0
733:            bra     s_rts
734: s_objE:
735:            lea.l   objE,a0
736:            bra     s_rts
737: s_objF:
738:            lea.l   objF,a0
739:            bra     s_rts
740:
741: s_rts:  MAGIC   __AUTO
742:            lea.l   pers,a0
743:            MAGIC   __AUTO
744:            rts
745:
746: game_over:
747:            lea.l   over_work,a1
748:            clr.w   46(a1)
749:
750:            moveq.l #40,d7
751:            move.w  #495,16(a1)
752: over2:
753:            bsr     over_sub
754:
755:            move.w  #10000,d5
756:            bsr     empty_loop
757:
758:            sub.w   #10,16(a1)
759:            add.w   #9,40(a1)
760:            dbf     d7,over2
761:
762:            clr.w   40(a1)
763:            add.w   #10,16(a1)
764: over3:
765:            bsr     over_sub
766:            subq.w  #4,46(a1)
767:            move.l  #5200,d5
768:            bsr     empty_loop
769:            bsr     ACHECK
770:            bne     over3
771:
772: over4:
773:            bsr     ACHECK
774:            beq     over4
775:
776:            bsr     cls
777:            lea.l   disp,a0
778:            MAGIC   __AUTO
779:            rts
780:
781: empty_loop:
782:            dbf     d5,empty_loop
783:            rts
784:
785: over_sub:
786:            movem.l d6-d7/a1,-(sp)
787:            lea.l   over_data,a0
788:            MAGIC   __3D_DATA
789:            movem.l (sp)+,d6-d7/a1
790:            move.l  a1,a0
791:            MAGIC   __AUTO
792:            rts
793:
794: cls:    move.b  #2,d1
795:            IOCS    _B_CLR_ST
796:            rts
797:
798: PAUSE:
799:            clr.w   d1
800:            IOCS    _BITSNS
801:            btst    #1,d0
802:            beq     PAUSE_RET
803: KEYCHECK2:
804:            clr.w   d1
805:            IOCS    _BITSNS
806:            btst    #1,d0
807:            bne     KEYCHECK2
808: KEYCHECK3:
809:            clr.w   d1
810:            IOCS    _BITSNS
811:            btst    #1,d0
812:            beq     KEYCHECK3
813: KEYCHECK4:
814:            clr.w   d1
815:            IOCS    _BITSNS
816:            btst    #1,d0
817:            bne     KEYCHECK4
818: PAUSE_RET:
819:            rts
820:
821: ACHECK:
822:            bsr     JOY
823:            btst    #5,d0
824:            rts
825:
826: TIMEPRINT:
827:            IOCS    _TIMEGET
828:            move.l  d0,d1
829:            IOCS    _TIMEBIN
830:            lea.l   TIME,a0
831:            move.l  (a0),d3
832:            move.l  d3,d4
833:            move.l  d0,d1
834:            sub.b   d3,d0
835:            tst.b d0
836:            bpl     TP2
837:            add.b   #60,d0
838:            sub.w   #$0100,d0
839: TP2:
840:            andi.b  #$00,d3
841:            sub.w   d3,d0
842:            tst.w   d0
843:            bpl     TP3
844:            add.w   #$3C00,d0
845:            sub.l   #$00010000,d0
846: TP3:
847:            andi.w  #$0000,d3
848:            sub.l   d3,d0
849:            tst.l   d0
850:            bpl     TP4
851:            add.l   #$180000,d0
852: TP4:
853:            andi.l  #$00FFFFFF,d0
854:            lea.l   TIMEPTR2,a1
855:            move.l  d0,d1
856:            IOCS    _TIMEASC
857:            move.w  #0,d1
858:            move.w  #18,d2
859:            IOCS    _B_LOCATE
860:            lea.l   TIMEPTR1,a1
861:            IOCS    _B_PRINT
862:            rts
863:
864: BREAK:
865:            move.w  #$c,d1
866:            IOCS    _BITSNS
867:            btst    #1,d0
868:            bne     RET
869:            rts
870: RET:
871:            IOCS    _OS_CURON
872:            IOCS    _B_CURON
873:
874:            DOS     _EXIT
875:
876:            .data
877:            .even
878:
879: initdata:
880:            dc.w    $12
881:            dc.w    $11
882:            dc.w    $101    *512*512 256
883:            dc.w    $7
884:            dc.w    $2
885:            dc.w    $9
886:            dc.w    $F
887: para:
888:            dc.w    $0b
889:            dc.w    0
890:            dc.w    0       * +4
891:
892:            dc.w    $0b
893:            dc.w    1
894:            dc.w    45      * +10
895:
896:            dc.w    $0b
897:            dc.w    2
898:            dc.w    10      * +16
899:
900:            dc.w    $0b
901:            dc.w    3
902:            dc.w    0       * +22
903:
904:            dc.w    $0b
905:            dc.w    4
906:            dc.w    0       * +28
907:
908:            dc.w    $0b
909:            dc.w    5
910:            dc.w    0       * +34
911:
912:            dc.w    $0b
913:            dc.w    6
914:            dc.w    0       * +40
915:
916:            dc.w    $0b
917:            dc.w    7
918:            dc.w    0       * +46
919:
920:            dc.w    $0b
921:            dc.w    8
922:            dc.w    0       * +52
923:            dc.w    $0f
924:
925: pers:   dc.w    $0d
926:            dc.w    $0F
927:
928: disp:   dc.w    $0E
929:            dc.w    $0F
930:
931: WAKU:   dc.w    $0c
932:            dc.w    28
933:            dc.w    -70,0,-70,-70,0,-50,-70,0,-30,-70,0,-10
934:            dc.w    -70,0,10,-70,0,30,-70,0,50,-70,0,70
935:            dc.w    70,0,-70,70,0,-50,70,0,-30,70,0,-10
936:            dc.w    70,0,10,70,0,30,70,0,50,70,0,70
937:            dc.w    -50,0,-70,-30,0,-70,-10,0,-70,10,0,-70
938:            dc.w    30,0,-70,50,0,-70,-50,0,70,-30,0,70
939:            dc.w    -10,0,70,10,0,70,30,0,70,50,0,70
940:            dc.w    16
941:            dc.w    $c0
942:            dc.w    0,8,1,9,2,10,3,11,4,12,5,13,6,14,7,15
943:            dc.w    0,7,8,15
944:            dc.w    16,22,17,23,18,24,19,25,20.26,21,27
945:            dc.w    $0F
946:
947: GOAL:   dc.w    $0c
948:            dc.w    4
949:            dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
950:            dc.w    4
951:            dc.w    $30
952:            dc.w    0,1,1,2,2,3,3,0
953:            dc.w    $0F
954: TRIANGLE:
955:            dc.w    $0c
956:            dc.w    3
957:            dc.w    0,5,6,-6,5,-6,6,5,-6
958:            dc.w    3
959:            dc.w    $c0
960:            dc.w    0,1,1,2,2,0
961:            dc.w    $0F
962:
963: obj1:   dc.w    $0c
964:            dc.w    8
965:            dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
966:            dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
967:            dc.w    5
968:            dc.w    $c0
969:            dc.w    0,1,0,4,1,5,2,6,3,7
970:            dc.w    $0F
971:
972: obj2:   dc.w    $0c
973:            dc.w    8
974:            dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
975:            dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
976:            dc.w    5
977:            dc.w    $c0
978:            dc.w    3,0,0,4,1,5,2,6,3,7
979:            dc.w    $0F
980:
981: obj3:   dc.w    $0c
982:            dc.w    8
983:            dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
984:            dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
985:            dc.w    6
986:            dc.w    $c0
987:            dc.w    0,1,3,0,0,4,1,5,2,6,3,7
988:            dc.w    $0F
989:
990: obj4:   dc.w    $0c
991:            dc.w    8
992:            dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
993:            dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
994:            dc.w    5
995:            dc.w    $c0
996:            dc.w    2,3,0,4,1,5,2,6,3,7
997:            dc.w    $0F
998:
999: obj5:   dc.w    $0c
1000:           dc.w    8
1001:           dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
1002:           dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
1003:           dc.w    6
1004:           dc.w    $c0
1005:           dc.w    0,1,2,3,0,4,1,5,2,6,3,7
1006:           dc.w    $0F
1007:
1008: obj6:  dc.w    $0c
1009:           dc.w    8
1010:           dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
1011:           dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
1012:           dc.w    6
1013:           dc.w    $c0
1014:           dc.w    2,3,3,0,0,4,1,5,2,6,3,7
```

▶12月号の表紙はいままででいちばん気に入ってます。水面の光具合といい（映り込み），皮膚の感触といいそそられますなあ。　井上　和也(22)福岡県

```
1015:           dc.w    $0F
1016:
1017: obj7:     dc.w    $0c
1018:           dc.w    8
1019:           dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
1020:           dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
1021:           dc.w    7
1022:           dc.w    $c0
1023:           dc.w    0,1,2,3,3,0,0,4,1,5,2,6,3,7
1024:           dc.w    $0F
1025:
1026: obj8:     dc.w    $0c
1027:           dc.w    8
1028:           dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
1029:           dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
1030:           dc.w    5
1031:           dc.w    $c0
1032:           dc.w    1,2,0,4,1,5,2,6,3,7
1033:           dc.w    $0F
1034:
1035: obj9:     dc.w    $0c
1036:           dc.w    8
1037:           dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
1038:           dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
1039:           dc.w    6
1040:           dc.w    $c0
1041:           dc.w    0,1,1,2,0,4,1,5,2,6,3,7
1042:           dc.w    $0F
1043:
1044: objA:     dc.w    $0c
1045:           dc.w    8
1046:           dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
1047:           dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
1048:           dc.w    6
1049:           dc.w    $c0
1050:           dc.w    1,2,3,0,0,4,1,5,2,6,3,7
1051:           dc.w    $0F
1052:
1053: objB:     dc.w    $0c
1054:           dc.w    8
1055:           dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
1056:           dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
1057:           dc.w    7
1058:           dc.w    $c0
1059:           dc.w    0,1,1,2,3,0,0,4,1,5,2,6,3,7
1060:           dc.w    $0F
1061:
1062: objC:     dc.w    $0c
1063:           dc.w    8
1064:           dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
1065:           dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
1066:           dc.w    6
1067:           dc.w    $c0
1068:           dc.w    1,2,2,3,0,4,1,5,2,6,3,7
1069:           dc.w    $0F
1070:
1071: objD:     dc.w    $0c
1072:           dc.w    8
1073:           dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
1074:           dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
1075:           dc.w    7
1076:           dc.w    $c0
1077:           dc.w    0,1,1,2,2,3,0,4,1,5,2,6,3,7
1078:           dc.w    $0F
1079:
1080: objE:     dc.w    $0c
1081:           dc.w    8
1082:           dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
1083:           dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
1084:           dc.w    7
1085:           dc.w    $c0
1086:           dc.w    1,2,2,3,3,0,0,4,1,5,2,6,3,7
1087:           dc.w    $0F
1088:
1089: objF:     dc.w    $0c
1090:           dc.w    8
1091:           dc.w    10,-5,10,-10,-5,10,-10,-5,-10,10,-5,-10
1092:           dc.w    10,5,10,-10,5,10,-10,5,-10,10,5,-10
1093:           dc.w    8
1094:           dc.w    $c0
1095:           dc.w    0,1,1,2,2,3,3,0,0,4,1,5,2,6
1096:           dc.w    $0F
1097:
1098: over_work:
1099:           dc.w    $0b
1100:           dc.w    0,0
1101:
1102:           dc.w    $0b
1103:           dc.w    1,0
1104:
1105:           dc.w    $0b
1106:           dc.w    2,0
1107:
1108:           dc.w    $0b
1109:           dc.w    3,0
1110:
1111:           dc.w    $0b
1112:           dc.w    4,0
1113:
1114:           dc.w    $0b
1115:           dc.w    5,0
1116:
1117:           dc.w    $0b
1118:           dc.w    6,0
1119:
1120:           dc.w    $0b
1121:           dc.w    7,0
1122:
1123:           dc.w    $0b
1124:           dc.w    8,0
1125:
1126:           dc.w    13
1127:           dc.w    14
1128:           dc.w    $F
1129: over_data:
1130:           dc.w    56
1131:           dc.w    -95,-15,0           *G
1132:           dc.w    -102,-20,0
1133:           dc.w    -108,-20,0
1134:           dc.w    -120,-10,0
1135:           dc.w    -120,10,0
1136:           dc.w    -108,20,0
1137:           dc.w    -95,10,0
1138:           dc.w    -95,20,0
1139:           dc.w    -95,0,0
1140:           dc.w    -108,0,0
1141:
1142:           dc.w    -90,20,0            *A
1143:           dc.w    -90,0,0
1144:           dc.w    -90,-10,0
1145:           dc.w    -78,-20,0
1146:           dc.w    -65,-10,0
1147:           dc.w    -65,0,0
1148:           dc.w    -65,20,0
1149:
1150:           dc.w    -60,20,0            *M
1151:           dc.w    -60,-20,0
1152:           dc.w    -48,-10,0
1153:           dc.w    -35,-20,0
1154:           dc.w    -35,20,0
1155:
1156:           dc.w    -5,-20,0            *E
1157:           dc.w    -30,-20,0
1158:           dc.w    -30,0,0
1159:           dc.w    -30,20,0
1160:           dc.w    -5,20,0
1161:           dc.w    -12,0,0
1162:
1163:           dc.w    5,15,0              *O
1164:           dc.w    12,20,0
1165:           dc.w    23,20,0
1166:           dc.w    30,15,0
1167:           dc.w    30,-15,0
1168:           dc.w    23,-20,0
1169:           dc.w    12,-20,0
1170:           dc.w    5,-15,0
1171:
1172:           dc.w    35,-20,0            *V
1173:           dc.w    35,5,0
1174:           dc.w    48,20,0
1175:           dc.w    60,5,0
1176:           dc.w    60,-20,0
1177:
1178:           dc.w    90,-20,0            *E
1179:           dc.w    65,-20,0
1180:           dc.w    65,0,0
1181:           dc.w    65,20,0
1182:           dc.w    90,20,0
1183:           dc.w    83,0,0
1184:
1185:           dc.w    95,20,0             *R
1186:           dc.w    95,0,0
1187:           dc.w    95,-20,0
1188:           dc.w    113,-20,0
1189:           dc.w    120,-15,0
1190:           dc.w    120,-5,0
1191:           dc.w    113,0,0
1192:           dc.w    120,10,0
1193:           dc.w    120,20,0
1194:
1195:           dc.w    45
1196:           dc.w    $30
1197:           dc.w    0,1,1,2,2,3,3,4,4,5,5,6,7,8,8,9
1198:           dc.w    10,12,12,13,13,14,14,16,11,15
1199:           dc.w    17,18,18,19,19,20,20,21
1200:           dc.w    22,23,23,25,25,26,24,27
1201:           dc.w    28,29,29,30,30,31,31,32,32,33,33,34,34,35,35,28
1202:           dc.w    36,37,37,38,38,39,39,40
1203:           dc.w    41,42,42,44,44,45,43,46
1204:           dc.w    47,49,49,50,50,51,51,52,52,53,53,54,54,55,48,53
1205:
1206: h:        ds.w    49
1207:
1208: MAZE:     dc.w    $0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0
,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0
  1209:           dc.w    $0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0
,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0
  1210:           dc.w    $0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0
,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0
  1211:           dc.w    $0,$0,$0,$D,$3,$0,$9,$5,$1,$5,$5,$5,$3,$0,$0,$9,$5,$5,$3
,$0,$9,$5,$3,$9,$5,$5,$1,$5,$5,$5,$1,$5,$5,$5,$5,$5,$7,$0,$0,$0
  1212:           dc.w    $0,$0,$0,$0,$C,$1,$2,$0,$A,$0,$0,$0,$C,$1,$5,$2,$0,$0,$C
,$1,$2,$B,$A,$E,$9,$5,$4,$5,$5,$3,$C,$5,$5,$5,$5,$5,$3,$0,$0,$0
  1213:           dc.w    $0,$0,$0,$B,$0,$8,$4,$1,$4,$5,$3,$0,$0,$A,$0,$8,$5,$3,$B
,$C,$2,$A,$A,$9,$6,$9,$5,$5,$5,$0,$5,$1,$5,$5,$5,$5,$6,$0,$0,$0
  1214:           dc.w    $0,$0,$0,$C,$5,$6,$0,$A,$9,$3,$C,$1,$1,$6,$0,$A,$B,$E,$8
,$3,$E,$A,$A,$E,$9,$2,$9,$5,$5,$4,$3,$C,$5,$5,$5,$5,$3,$0,$0,$0
  1215:           dc.w    $0,$0,$0,$0,$0,$0,$0,$A,$C,$4,$3,$C,$2,$0,$0,$A,$8,$5,$6
,$8,$5,$6,$A,$9,$4,$6,$A,$9,$1,$3,$C,$5,$5,$3,$9,$3,$A,$0,$0,$0
  1216:           dc.w    $0,$0,$0,$D,$5,$5,$1,$4,$3,$0,$B,$7,$A,$0,$0,$A,$E,$9,$3
,$A,$9,$1,$2,$A,$D,$5,$6,$8,$4,$0,$1,$3,$9,$4,$4,$6,$A,$0,$0,$0
  1217:           dc.w    $0,$0,$0,$0,$0,$0,$A,$0,$A,$D,$0,$1,$2,$0,$9,$4,$1,$4,$2
,$A,$C,$4,$6,$8,$1,$5,$5,$2,$0,$C,$0,$6,$E,$9,$1,$3,$A,$0,$0,$0
  1218:           dc.w    $0,$0,$0,$D,$3,$0,$C,$1,$6,$0,$A,$C,$2,$0,$A,$0,$A,$0,$A
,$C,$1,$5,$5,$4,$6,$0,$0,$E,$0,$0,$E,$9,$5,$0,$0,$2,$A,$0,$0,$0
  1219:           dc.w    $0,$0,$0,$0,$A,$0,$0,$A,$D,$1,$0,$3,$A,$0,$A,$0,$A,$0,$C
,$3,$C,$5,$1,$5,$1,$1,$5,$5,$3,$0,$0,$A,$B,$C,$4,$6,$A,$0,$0,$0
  1220:           dc.w    $0,$0,$0,$0,$A,$0,$0,$8,$7,$8,$4,$2,$8,$1,$0,$5,$2,$0,$0
,$8,$1,$7,$C,$7,$A,$E,$D,$3,$C,$3,$0,$A,$C,$5,$1,$7,$A,$0,$0,$0
  1221:           dc.w    $0,$0,$0,$0,$A,$0,$9,$6,$9,$6,$B,$E,$8,$4,$6,$0,$C,$1,$1
,$4,$2,$D,$5,$5,$2,$0,$9,$6,$0,$C,$1,$0,$5,$3,$C,$3,$A,$0,$0,$0
  1222:           dc.w    $0,$0,$0,$0,$A,$0,$A,$9,$6,$9,$4,$1,$6,$0,$0,$B,$0,$8,$6
,$0,$C,$3,$0,$0,$8,$5,$2,$B,$B,$0,$8,$6,$B,$C,$3,$C,$2,$0,$0,$0
  1222:           dc.w    $0,$0,$0,$0,$A,$0,$A,$9,$6,$9,$4,$1,$6,$0,$0,$B,$0,$8,$6
,$0,$C,$3,$0,$0,$8,$5,$2,$B,$B,$0,$8,$6,$B,$C,$3,$C,$2,$0,$0,$0
  1223:           dc.w    $0,$0,$0,$0,$8,$5,$6,$E,$9,$6,$0,$A,$0,$0,$9,$4,$3,$A,$0
,$0,$0,$8,$3,$9,$6,$0,$A,$A,$C,$5,$6,$D,$4,$3,$C,$7,$A,$0,$0,$0
  1224:           dc.w    $0,$0,$0,$0,$A,$0,$0,$9,$6,$0,$0,$A,$D,$5,$2,$0,$A,$8,$3
,$0,$0,$8,$6,$C,$5,$7,$A,$C,$5,$1,$1,$5,$5,$4,$1,$1,$2,$0,$0,$0
  1225:           dc.w    $0,$0,$0,$0,$C,$3,$0,$E,$0,$0,$9,$4,$3,$0,$A,$9,$6,$8,$6
,$0,$D,$2,$0,$D,$1,$7,$8,$5,$3,$A,$A,$9,$5,$3,$A,$A,$A,$0,$0,$0
  1226:           dc.w    $0,$0,$0,$0,$0,$A,$0,$0,$0,$9,$2,$0,$E,$0,$E,$8,$5,$6,$0
,$B,$0,$A,$0,$9,$0,$3,$C,$7,$A,$A,$A,$E,$8,$5,$2,$A,$A,$0,$0,$0
  1228:           dc.w    $0,$0,$0,$A,$0,$A,$0,$0,$A,$0,$A,$0,$9,$5,$5,$2,$0,$B,$0
,$8,$1,$4,$3,$8,$0,$0,$5,$5,$6,$A,$C,$5,$6,$9,$6,$A,$A,$0,$0,$0
  1229:           dc.w    $0,$0,$0,$A,$0,$C,$5,$3,$C,$1,$4,$5,$6,$9,$3,$C,$5,$0,$5
,$4,$2,$0,$A,$8,$4,$6,$9,$7,$9,$0,$3,$9,$7,$A,$9,$2,$A,$0,$0,$0
  1230:           dc.w    $0,$0,$0,$E,$0,$0,$0,$E,$D,$6,$D,$5,$1,$2,$C,$5,$7,$A,$0
,$0,$C,$5,$6,$A,$0,$D,$2,$0,$A,$A,$A,$A,$9,$2,$A,$A,$A,$0,$0,$0
  1231:           dc.w    $0,$0,$0,$B,$9,$5,$5,$5,$5,$5,$5,$3,$C,$2,$9,$5,$5,$0,$1
,$3,$0,$0,$0,$C,$5,$3,$A,$0,$A,$A,$A,$C,$6,$A,$A,$A,$A,$0,$0,$0
  1232:           dc.w    $0,$0,$0,$C,$6,$0,$0,$0,$0,$B,$0,$8,$3,$A,$A,$9,$3,$8,$0
,$0,$5,$5,$3,$0,$0,$A,$A,$0,$A,$A,$8,$5,$3,$A,$A,$A,$A,$0,$0,$0
  1233:           dc.w    $0,$0,$0,$0,$B,$D,$1,$5,$5,$0,$7,$8,$4,$6,$A,$A,$A,$C,$4
,$6,$D,$3,$A,$9,$1,$2,$C,$1,$6,$A,$A,$9,$6,$A,$A,$A,$A,$0,$0,$0
  1234:           dc.w    $0,$0,$0,$D,$0,$7,$8,$1,$3,$E,$0,$A,$0,$0,$A,$A,$8,$5,$5
,$5,$5,$4,$4,$4,$4,$6,$9,$6,$B,$A,$A,$A,$B,$A,$A,$E,$A,$0,$0,$0
  1235:           dc.w    $0,$0,$0,$0,$C,$3,$C,$0,$4,$1,$1,$4,$1,$5,$2,$E,$A,$9,$5
,$3,$0,$9,$5,$1,$3,$9,$2,$9,$6,$A,$A,$A,$A,$A,$C,$3,$A,$0,$0,$0
  1236:           dc.w    $0,$0,$0,$9,$1,$0,$7,$A,$0,$B,$6,$0,$A,$0,$A,$9,$6,$A,$0
,$A,$D,$6,$9,$6,$B,$4,$2,$8,$5,$2,$8,$2,$A,$C,$5,$6,$A,$0,$0,$0
  1237:           dc.w    $0,$0,$0,$8,$4,$6,$9,$2,$0,$A,$0,$9,$2,$0,$A,$A,$9,$6,$0
,$E,$D,$5,$2,$9,$6,$9,$6,$A,$9,$2,$A,$E,$C,$5,$5,$3,$A,$0,$0,$0
  1238:           dc.w    $0,$0,$0,$A,$D,$5,$0,$6,$0,$A,$0,$8,$6,$0,$A,$A,$A,$0,$B
,$9,$5,$5,$6,$A,$9,$2,$9,$2,$A,$A,$E,$9,$5,$5,$5,$6,$A,$0,$0,$0
  1239:           dc.w    $0,$0,$0,$C,$1,$3,$A,$0,$0,$A,$0,$A,$0,$0,$A,$A,$A,$0,$A
,$E,$9,$1,$3,$A,$A,$A,$A,$A,$A,$C,$5,$4,$1,$5,$5,$3,$A,$0,$0,$C
  1240:           dc.w    $0,$0,$0,$0,$C,$2,$C,$5,$1,$6,$0,$A,$0,$D,$2,$A,$A,$0,$C
,$1,$4,$4,$2,$A,$A,$E,$E,$E,$C,$5,$5,$7,$C,$5,$7,$A,$A,$0,$0,$0
  1241:           dc.w    $0,$0,$0,$B,$0,$A,$0,$0,$A,$0,$0,$C,$3,$0,$A,$E,$A,$0,$0
,$A,$0,$0,$A,$E,$8,$5,$5,$5,$5,$5,$1,$5,$1,$5,$5,$2,$A,$0,$0,$0
  1242:           dc.w    $0,$0,$0,$C,$1,$4,$3,$0,$C,$5,$3,$0,$A,$0,$8,$5,$4,$1,$5
,$4,$5,$5,$6,$B,$A,$D,$5,$3,$9,$3,$A,$9,$4,$5,$3,$A,$A,$0,$0,$0
  1243:           dc.w    $0,$0,$0,$0,$A,$0,$8,$3,$0,$0,$A,$0,$E,$0,$A,$0,$0,$A,$0
,$D,$5,$5,$5,$0,$4,$5,$5,$2,$A,$E,$A,$A,$9,$5,$6,$2,$E,$0,$0,$0
  1244:           dc.w    $0,$0,$0,$D,$6,$0,$C,$4,$5,$5,$6,$0,$D,$5,$4,$7,$0,$C,$5
,$5,$5,$5,$7,$E,$D,$5,$5,$6,$C,$5,$4,$4,$4,$5,$7,$C,$7,$0,$0,$0
  1245:           dc.w    $0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0
,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0
  1246:           dc.w    $0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0
,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0
  1247:           dc.w    $0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0
,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0,$0
  1248:
  1249: K:        ds.w    8
  1250: TIME:     ds.l    1
  1251: TIMEPTR1:
  1252:           dc.b    "TIME:"
  1253: TIMEPTR2:
  1254:           ds.b    10
  1255: OVERPTR:
  1256:           dc.b    "GAME  OVER",13,10,0
  1257: TITLEPTR:
  1258:           dc.b    "                     MAZE  Original APRO SEP.1991",13,10,1
3,10
  1259:           dc.b    "                        PUSH A BUTTON TO START",13,10,0
  1260: CLEARPTR:
  1261:           dc.b    "                         CONGRATULATIONS!",13,10,0
  1262:
  1263:           .end
```

# マシン語カクテル in Z80's Bar

## 第28回——新年会模擬パーティ——

シナリオ：**金子俊一**

**いまの時期には宴会がつきもの。というわけで，やっぱりここでも得体のしれないパーティが行われているようです。しかも，宴会をしながら，宴会のシミュレーションを作り，実行させるというわけのわからないことを始めたようなのですが……。**

♪カラン，コロ～ン

**源光**（以下光）：こんにちは。

**ようこ**（以下Yo）：ハッピー，ニューイヤー！

**メアリー**（以下メ）：A happy New Year！

**西川善司**（以下善）：マッピー，ニュージーランドストーリー！

**長老**（以下老）：ほっほっほっほ。

**古村聡**（以下で）：もしゃもしゃ……。

**山田純二**（以下純）：やっほ～！

**柴田淳**（以下Ats）：どーもー。あっちゃんどぇ～す。

**光**：みんな，すっかりできあがっちゃってますね。

**マスター**（以下M）：だろう。ようこちゃんが新年会やろうっていいだしたんだよ。

**Yo**：なによ，私が悪いとでもいいたいの？

**光**：完全にからみ酒ですね。

**メ**：Hi，ひかる。おひっさしぶりね。

**光**：あれ，いつ日本に戻ってきたの？

**メ**：それ，yesterdayね。

**光**：すっかり日本語下手になったね。

**メ**：Don't worry. タウンページで調べてね。

**善**：なにいってんだか。僕が通訳してやろう。

**Yo**：えゞっ。善ちゃん，英語できるの？

**善**：失礼な，これでも私はイギリスへ留学してたんですぞ。

**M**：人は見掛けによらないってのは，このことですねえ。

**光**：それにしても本当にフルメンバーですね。

**Yo**：新年会をやりたくって，私が呼んだのよ。

### さよならくんコース

**M**：でもねえ，うちの店でこんなにいっぱい入っちゃうと，パーティーどころの騒ぎじゃないと思うんだけど。

**Yo**：商売繁盛で，ササ持ってこいってね。

**で**：マスター，焼きウドン追加ね。

**M**：はいはい。

**純**：僕はメッコールとあたりめ。

**メ**：サスケ，please.

**老**：スモーキー&カンパニーのミント。

**善**：ユンケル・ゴールド。

**Ats**：ほろにが。

**M**：どっか～ん。

**光**：あ～あ，マスターが「さよならくん」になっちゃった。

**Yo**：ちょっと注文が多かったかしらね。

**光**：ちょっとこの人数では無理ないかもしれませんね。

**老**：いや，やってみなくてはわからんぞ。

**純**：コンピュータにやらせたら？

**Ats**：どーいうことですか。

**で**：そんなの決まってんじゃん。シュミレーションをするんだよ。もぐもぐ。

**純**：それをいうならシミュレーション。

**Ats**：さっすが山田さん。だてに編集のバイトをやってませんね。

**純**：バイトじゃないの。正社員なの。

### 誰が作るのプログラム

**光**：ということで，いいだしっぺの純ちゃんがプログラムを作るっていうことで。

**純**：なにを寝ぼけてるの？

**Yo**：ネボちゃんなの？

**純**：私は編集なの。プログラムはあんまり作らないの。

**で**：それじゃあ，クジで決めたら？ モグモグ。

**光**：いや，やめておきましょう。僕のクジ運の悪さは天下一品ですから。年末ジャンボ宝くじも300円以上は当たったことがないし。

**Yo**：そういえば，こういった場面でプログラムを組むのはたいてい光君だわね。

**老**：ふぉっふぉっふぉ。おぬし，負けを認めたな。さあ，今夜の酒代じゃ。プログラムをた～んと作っておくれ。

**光**：ふっふっふ。僕には強力な味方がいるんだ。さあ，柴田君，フォーメーションBだ！

**純**：なに寝言いってんの。

**Yo**：ネゴちゃんなの？

**純**：柴田君は「ほろにが」飲んで酔っぱらっちゃってるよ。

**Yo**：ほんとだ。お店の中をぐるぐる駆け回ってる。

**光**：だめだ，こりゃ。

### 行動を設定しよう

**光**：それじゃあ観念してプログラムでも作りますか。

**メ**：fightするあるよ。

**光**：えっと，パーティ会場は40×25のマス目全部としよう。

**Yo**：テーブルも置かなくちゃね。

**善**：いすはないの？

**光**：まあ，立食パーティということで。

で：立ち食いソバとか立ち食いウドン？
光：私は「りっしょくパーティ」といったんだ。「たちぐいパーティ」なんかじゃないわい。
Yo：食べ物も用意しなくちゃね。
老：それはテーブルに置いてあることにすればいいじゃろう。
光：それから全員の行動パターンを決めておかなくちゃな。

＊　＊　＊

・マスターの場合
全員の注文をさばくのに精一杯。いつも厨房にいて，姿は見せない。厨房の座標は(0,0)の上あたりにあることにする。

・ようこの場合
注文取りに駆けずり回る。全員の注文を取り終わったら，マスターのいる厨房に報告しにいく。それが終わったら，すぐにまた注文取りにいく。

・ひかるの場合
ようこをひたすら追い掛ける。ただし，テーブルを迂回していくというような知恵はない。ひたすら純情に直線的に，ようこのところへ向かっていく。

・こむらの場合
とりあえず，食べる。ひたすら食べる。テーブルにへばりつき，片っ端から食べていく。無意識のうちに時計回りをしている。もちろん，食ったら寝るのはウシの基本。

・あつしの場合
もう酔っぱらっている。ひたすらお店の中を駆けずり回る。これは時計逆回り。

・ぜんじの場合
酔っている。完全に酔歩。ランダムウオークともいう。ときどき居眠りをこく。要するにネボちゃん。

・メアリーの場合
話し友達がほしい。そこで，英語を話せるぜんじを追い掛ける。ぜんじが寝ていてもそこで待っている。

・長老の場合
ぜんじの師匠。基本的に行動パターンは同じで，ランダムウオークか居眠り。年寄りのせいか，ぜんじより眠りやすいが，眠りも浅くて起きやすい。

・じゅんじの場合
編集者。立場を考えて，みんなの和を保とうとする。よって，全員の中心にいようと心掛ける。

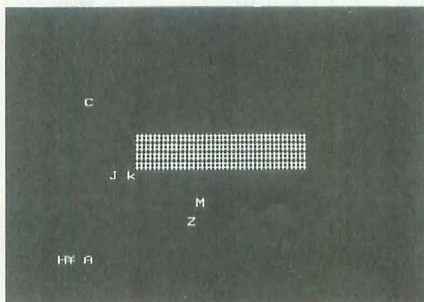
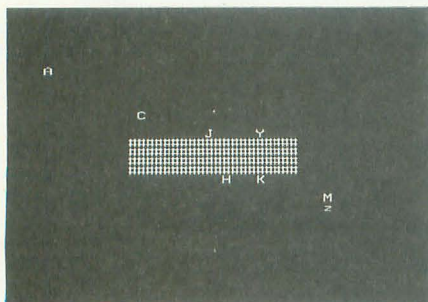
・うりゃかわの場合
特別監修。なにをするわけでもなく，画面にも出てこない。ただし，原稿を監修しているので，編集者であるじゅんじの行動に影響を与える。

＊　＊　＊

光：こんなとこかな。
Yo：リアリティがあるような，ないような感じね。
で：ちっともリアルじゃないよ。モグモグ。食うか寝るかのどっちかだけなんてことはありえないもの。ぐぅ～。
純：寝ちゃったよ。
Yo：やっぱりリアルね。

## やっとプログラム

光：それではプログラムを組みましょう。
Yo：光君がんばってね。私たちは勝手に騒いでるから。
光：そんなあ。
老：ほっほっほ。カタなしじゃのう。
光：しょうがないなあ，カチャカチャカチャ……。できたっと。
M：どれどれ？
光：JA000で実行。スペースキーで一時停止，ブレイクキーでS-OSに戻ります。
Yo：S-OS用なのね。
M：それはポイント高いですね。
光：ウエイトルーチンの設定をいじくれば，遅いマシンでも大丈夫です。

老：ほっほっほ。面白いのう。
M：「ポピュラス」みたいにキャラクターに寿命があってもいいのに。
老：縁起でもないことをぬかすのう，マスター。
光：まあまあ，マスターったら自分のキャラが出てこないんで，すねてるんですよ。
純：さて，プログラムの解説だ。
光：えっとですねえ，いろいろやってるんですよ。
Ats：それじゃあ解説になってないよぉ……（ドップラー効果）。
Yo：まだ走ってる。そういえば，シミュレーションでは“あつし”より“こむら”の動きのほうが遅いようにみえるけど。
光：うん，遅いよ。
Yo：どうして？
光：今回のキャラたちはたいていそうなんだけど，動くときにいろいろな条件があるんだ。たとえば，AWALK，HWALK，JWALKなんていうワークエリアがあるでしょ。
Yo：うんうん。
光：そこの値が，$AAとか$F0とかになってるよね。
Yo：うん。
光：この値をシフトしていって，ビットの状態を見ながら，今回はX座標，今回はY座標，今回はランダムといった具合に，簡単な8パターンの繰り返しをさせているんだ。
メ：よくわかんなーす。
善：なんだかちっともわからないって，メアリーがいってるよ。
老：いまのは日本語だったぞ。
光：それじゃあね，下位1ビットが1だったらキャラを動かして，0だったら動かさないとするでしょ。
メ：ふむ踏む。
善：へんなとこに漢字を使わなくてもいいんだぞ，メアリー。
光：そこで，1010_1010（$AA）というデータをシフトしてやると，0101_0101（$55）になって，下位1ビットを見ると1になっているからキャラを動かすことになる。
老：もう1度シフトすると1010_1010（$AA）になり，下位1ビットは0だから動かさないようになるわけじゃな。
光：そうです。その，1010_1010とかの値に

0を多めに入れておくと動きは鈍くなるし，1が多いならせわしなく動くことになる。
メ：なるへそ。
光：そこで，1010_1011（$AB）とかして，下位3ビットを見て，全部1なら寝るとかにしておくと，8回に1回だけの指示とかができるようになる。この場合，

| | | |
|---|---|---|
| 1010_1011 | ：動く | |
| 0101_0111 | ：寝る | * |
| 1010_1110 | ：動かない | |
| 0101_1101 | ：動く | |
| 1011_1010 | ：動かない | |
| 0111_0101 | ：動く | |
| 1110_1010 | ：動かない | |
| 1101_0101 | ：動く | |
| 1010_1011 | ：元に戻る | |

という，パターンを指示しているわけだ。
Yo：なるほど。
純：話は変わりますが手を抜きましたね，乱数。
光：はあっはっは。ばれちゃあしょうがねえ。最初に乱数のサブルーチンで苦労したんですよ。そこで，メモリを片っ端から持ってきちゃえ，と。
純：ってことは，メモリにはなにか入っていたほうがランダムに動くわけですね。
光：そうね。$00とか$FFばっかしじゃあ，ハマリのパターンになっちゃうかも。とりあえず，なにかプログラムなり，データなりがメモリに散らばっているほうが面白いはずです。
老：最悪の場合はどうなるのかのう。
光：そうですね。$00ばっかりだと，X座標は左に，Y座標なら上に集まりやすいでしょうね。
Yo：$FFばっかりだと？
光：居眠りする連中がずっと起きなくなる可能性が大です。
老：このキャラクターが小文字になっているのが寝ている状態というわけじゃな。
光：ええ。
Yo：私の“¥”マークはなに？
光：注文を取ったときと，マスターに報告したときです。
Yo：なんだか金の亡者みたいでいやだわ。
光：最初は小文字の“y”だったんですけどね，なんだか蹴っとばしてるように見えるんですよ。
老：寝てる人とも区別がつかないしのう。

純：小文字がない機種の人はどうすればいいの？
光：う〜ん。もうしわけないけど，ほかのキャラクタでそれらしいものを代わりに使うしかないなあ。
老：それではほれ，10月号に載っておったX1用のZ80'sBarとかいうゲームでPCG定義されてたじゃろ。あれは使えんかのう。

光：多少の改造が必要ですが，基本的な部分はどうにかなるんじゃないのかな。それにキャラが何人か足りないから，自分で作るしかないでしょうね。
Yo：それはにぎやかになりそうね。
老：たしか，ようこちゃんのキャラもなかったように思うのじゃが。
Yo：ゆー，るー，さーん。
純：そういえば乱数以外でハマリパターンっていうのはあるの？
光：えっと，初期配置は乱数で決まるんだけど，まるっきり融通のきかない“こむら”と“あつし”が隣り合った場合は動かなくなるかもしれない。画面の上半分では“KA”の順に並んでいたらハマリ。下半分では“AK”の順に並んでいたらハマリ。
老：時計回りとその逆回りしかプログラムされてないわけじゃな。
光：ええ，いろいろ試行錯誤して，ランダムを少しでも入れておくとハマリは少ないことがわかったんですが，それだけじゃつまらないから。
純：あとはあるの？
光：テーブルにへばりつきっぱなしの“こむら”や“じゅんじ”を“ようこ”が追い掛けてるときも可能性があります。
老：誰もがテーブルを苦手としているようじゃな。
光：そうですね。

## これからどうなる

で：改良方法とかはないの？
光：行動パターンを増やすと面白いかもね。
で：たとえば？
光：えっと，最初の何ターンかは全員がテーブルについていて，酔っぱらいだすと動き回り始めるとか。
Yo：ちゃんと出入り口を決めておいて適当に集まってきて，適当に帰っていくとかもいいかも。
M：営業時間を決めておいて，それまではようこちゃんと私で準備しているとか。
Yo：営業時間が終わったら掃除もしなくちゃね，マスター。
純：キャラが隣り合ってしゃべっているときは「ふきだし」をつけてもいいんじゃない？
善：いっそのこと画面下の数行を会話用に空けておいて，相手によって会話するパターンが変わるとか。
光：みんな好き勝手いいやがって。
Ats：でも面白そうですね。
光：プログラムしてて思ったんだけど，自分で組んだほうが絶対に楽しいよ。
Yo：そう？
光：最初のうちなんか，“ぜんじ”と“こむら”と“あつし”しか動いてなくって，なんだか「A列車で行こう」をプログラムしているみたいだったし。
老：ほうほう。2人が決まった線路の上を走っているようなもんじゃからなぁ。
光：僕が作ったプログラムを参考に，自分なりの新年会を開くといいかもね。

## 今日はおごりだ

光：僕ってとことんタダ酒に縁がないんだなあ。
M：そんなことはないですよ。今日のとこ

ろは私のおごりということにしておきましょう。

**光**：やった～。

**M**：でもツケがあるから，プログラムはいただいておくよ。

**光**：がっくし。

**Yo**：しょうがないわね。

**光**：では気を取り直して，新年のご挨拶。

**一同**：本年もよろしくお願いします。

**善**：いたしま～す。

**老**：タラちゃんか，おぬしは。

—つづく—

## リスト1

```
0000                 1 ;      PARTY SIMULATION
0000                 2 ;
0000                 3 ;          by Hikaru Minamoto
0000                 4
A000                 5           ORG      $A000
A000                 6
A000                 7 ;Label          Address   Break
A000                 8
A000                 9
A000                10 #WIDTH  EQU     $1F5C    ; system work
A000                11 #PAUSE  EQU     $1FC7    ; AF
A000                12 #PRINTS EQU     $1FF1    ; F
A000                13 #PRINT  EQU     $1FF4    ; F
A000                14 #SCRN   EQU     $201B    ; AF
A000                15 #LOC    EQU     $201E    ; AF
A000                16 #WIDCH  EQU     $2030    ; AF,BC,DE,HL
A000                17
A000                18
A000                19 BEGIN
A000 3A 5C 1F       20         LD      A,(#WIDTH)
A003 32 E9 A3       21         LD      (@WIDTH),A
A006                22         ;
A006 3E 28          23         LD      A,40
A008 CD 30 20       24         CALL    #WIDCH
A00B                25         ;
A00B                26 TABLE
A00B 21 0A 0A       27         LD      HL,$0A0A          ; H=Y=10,L=X=10
A00E                28 TABL2
A00E CD 1E 20       29         CALL    #LOC
A011 06 14          30         LD      B,20
A013 3E 23          31         LD      A,"#"
A015                32 TABL3
A015 CD F4 1F       33         CALL    #PRINT
A018 10 FB          34         DJNZ    TABL3
A01A 24             35         INC     H
A01B 3E 0D          36         LD      A,13
A01D BC             37         CP      H
A01E 30 EE          38         JR      NC,TABL2
A020                39         ;
A020                40 SETCHR
A020 AF             41         XOR     A
A021 32 7D A2       42         LD      (ZSLEEP),A
A024 32 B4 A2       43         LD      (CSLEEP),A
A027 32 99 A3       44         LD      (KSLEEP),A
A02A 32 95 A1       45         LD      (ORDER),A
A02D                46         ;
A02D CD 94 A0       47         CALL    INIT     ;YOUKO
A030 CD 8D A1       48         CALL    YOUPR
A033                49         ;
A033 CD 94 A0       50         CALL    INIT     ;HIKARU
A036 CD D1 A1       51         CALL    HIKPR
A039                52         ;
A039 CD 94 A0       53         CALL    INIT     ;MEARY
A03C CD 3E A2       54         CALL    MEAPR
A03F                55         ;
A03F CD 94 A0       56         CALL    INIT     ;ZENJI
A042 CD 75 A2       57         CALL    ZENPR
A045                58         ;
A045 CD 94 A0       59         CALL    INIT     ;CHOUROU
A048 CD AC A2       60         CALL    CHOPR
A04B                61         ;
A04B CD 94 A0       62         CALL    INIT     ;JUNJI
A04E CD F1 A2       63         CALL    JUNPR
A051                64         ;
A051 CD 94 A0       65         CALL    INIT     ;KOMURA
A054 CD 91 A3       66         CALL    KOMPR
A057                67         ;
A057 CD 94 A0       68         CALL    INIT     ; ATUSI
A05A CD DE A3       69         CALL    ATUPR
A05D                70         ;
A05D                71 MAIN
A05D 00 00 00       72         DS      3        ; DUMMY
A060                73         ;
A060 CD 94 A0       74         CALL    INIT     ; URAKAWA
A063 2A EA A3       75         LD      HL,(@LOC)
A066 22 EC A3       76         LD      (ULOC),HL
A069                77         ;
A069 CD 0A A1       78         CALL    YOUKO
A06C CD 97 A1       79         CALL    HIKARU
A06F CD 02 A2       80         CALL    MEARY
A072 CD 47 A2       81         CALL    ZENJI
A075 CD 7E A2       82         CALL    CHOUROU
A078 CD B5 A2       83         CALL    JUNJI
A07B CD 1A A3       84         CALL    KOMURA
A07E CD 9B A3       85         CALL    ATUSI
A081 CD E7 A3       86         CALL    MASTER   ; DUMMY
A084 CD A5 A0       87         CALL    WAIT
A087                88         ;
A087 CD C7 1F       89         CALL    #PAUSE
A08A 8E A0          90         DW      END
A08C 18 CF          91         JR      MAIN
A08E                92 END
A08E 3A E9 A3       93         LD      A,(@WIDTH)
A091 C3 30 20       94         JP      #WIDCH
A094                95         ;
A094                96 INIT
A094 CD D6 A0       97         CALL    RND
A097 E6 27          98         AND     39
A099 32 EA A3       99         LD      (@LOC),A
A09C CD D6 A0      100         CALL    RND
A09F E6 17         101         AND     23
A0A1 32 EB A3      102         LD      (@LOC+1),A
A0A4 C9            103         RET
A0A5               104         ;
A0A5               105 WAIT
A0A5 C5            106         PUSH    BC
A0A6 01 00 A0      107         LD      BC,$A000
A0A9 0B            108         DEC     BC
A0AA 79            109         LD      A,C
A0AB B0            110         OR      B
A0AC 20 FB         111         JR      NZ,WAIT+4
A0AE C1            112         POP     BC
A0AF C9            113         RET
A0B0               114         ;
A0B0               115 CHRPR
A0B0 F5            116         PUSH    AF
A0B1 2A EA A3      117         LD      HL,(@LOC)
A0B4 CD 1B 20      118         CALL    #SCRN
A0B7 FE 20         119         CP      " "
A0B9 20 06         120         JR      NZ,CHRPR2
A0BB EB            121         EX      DE,HL
A0BC 73            122         LD      (HL),E   ; LD (?LOC),HL
A0BD 23            123         INC     HL
A0BE 72            124         LD      (HL),D
A0BF 2B            125         DEC     HL
A0C0 EB            126         EX      DE,HL
A0C1               127 CHRPR2
A0C1 EB            128         EX      DE,HL
A0C2 5E            129         LD      E,(HL)
A0C3 23            130         INC     HL
A0C4 56            131         LD      D,(HL)
A0C5 EB            132         EX      DE,HL
A0C6 CD 1E 20      133         CALL    #LOC
A0C9 F1            134         POP     AF
A0CA C3 F4 1F      135         JP      #PRINT
A0CD               136         ;
A0CD               137 CHRDEL
A0CD 22 EA A3      138         LD      (@LOC),HL
A0D0 CD 1E 20      139         CALL    #LOC
A0D3 C3 F1 1F      140         JP      #PRINTS
A0D6               141         ;
A0D6               142 RND
A0D6 E5            143         PUSH    HL
A0D7               144 RND2    ;
A0D7 2A E1 A0      145         LD      HL,(RNDADR)
A0DA 23            146         INC     HL
A0DB 22 E1 A0      147         LD      (RNDADR),HL
A0DE 7E            148         LD      A,(HL)
A0DF               149         ;
A0DF E1            150         POP     HL
A0E0 C9            151         RET
A0E1               152 RNDADR
A0E1 00 00         153         DS      2
A0E3               154         ;
A0E3               155 RNDXY
A0E3 CD D6 A0      156         CALL    RND
A0E6 E6 01         157         AND     1
A0E8 47            158         LD      B,A
A0E9 EE 01         159         XOR     1
A0EB 4F            160         LD      C,A
A0EC               161         ;
A0EC 7E            162         LD      A,(HL)
A0ED 80            163         ADD     A,B
A0EE 91            164         SUB     C
A0EF FE 27         165         CP      39
A0F1 30 03         166         JR      NC,RNDY
A0F3 32 EA A3      167         LD      (@LOC),A
A0F6               168 RNDY
A0F6 CD D6 A0      169         CALL    RND
A0F9 E6 01         170         AND     1
A0FB 47            171         LD      B,A
A0FC EE 01         172         XOR     1
A0FE 4F            173         LD      C,A
A0FF               174         ;
A0FF 23            175         INC     HL
A100 7E            176         LD      A,(HL)
A101 80            177         ADD     A,B
A102 91            178         SUB     C
A103 FE 17         179         CP      23
A105 D0            180         RET     NC
A106 32 EB A3      181         LD      (@LOC+1),A
A109 C9            182         RET
A10A               183         ;
A10A               184 YOUKO
A10A 00 00 00      185         DS      3        ; DUMMY
A10D               186         ;
A10D 2A EE A3      187         LD      HL,(YLOC)
A110 CD CD A0      188         CALL    CHRDEL
A113               189         ;
A113 3A 96 A1      190         LD      A,(YOUWALK)
A116 07            191         RLCA
A117 32 96 A1      192         LD      (YOUWALK),A
A11A E6 07         193         AND     7
A11C 28 3C         194         JR      Z,YOURND
A11E E6 01         195         AND     1
A120 28 1C         196         JR      Z,YOUKOY
A122               197 YOUKOX
A122 21 F0 A3      198         LD      HL,HLOC
A125 3A 95 A1      199         LD      A,(ORDER)
A128 5F            200         LD      E,A
A129 16 00         201         LD      D,0
A12B 19            202         ADD     HL,DE
A12C 6E            203         LD      L,(HL)   ; LD L,(HLOC+ORDER)
A12D 3A EE A3      204         LD      A,(YLOC)
A130 BD            205         CP      L
A131 28 0B         206         JR      Z,YOUKOY
A133 30 03         207         JR      NC,YOUKO2
A135 3C            208         INC     A
```

```
A136 18 01      209          JR      YOUKO2+1
A138            210 YOUKO2
A138 3D         211          DEC     A
A139 32 EA A3   212          LD      (@LOC),A
A13C 18 22      213          JR      YORDER
A13E            214          ;
A13E            215 YOUKOY
A13E 21 F1 A3   216          LD      HL,HLOC+1
A141 3A 95 A1   217          LD      A,(ORDER)
A144 5F         218          LD      E,A
A145 16 00      219          LD      D,0
A147 19         220          ADD     HL,DE
A148 6E         221          LD      L,(HL)  ; LD L,(HLOC+1+ORDER)
A149 3A EF A3   222          LD      A,(YLOC+1)
A14C BD         223          CP      L
A14D 28 D3      224          JR      Z,YOUKOX
A14F 30 03      225          JR      NC,YOUKO3
A151 3C         226          INC     A
A152 18 01      227          JR      YOUKO3+1
A154            228 YOUKO3
A154 3D         229          DEC     A
A155 32 EB A3   230          LD      (@LOC+1),A
A158 18 06      231          JR      YORDER
A15A            232 YOURND
A15A 21 EE A3   233          LD      HL,YLOC
A15D CD E3 A0   234          CALL    RNDXY
A160            235 YORDER
A160 21 F0 A3   236          LD      HL,HLOC
A163 3A 95 A1   237          LD      A,(ORDER)
A166 5F         238          LD      E,A
A167 16 00      239          LD      D,0
A169 19         240          ADD     HL,DE
A16A 5E         241          LD      E,(HL)
A16B 23         242          INC     HL
A16C 56         243          LD      D,(HL)  ; DE=(HLOC+ORDER)
A16D 2A EA A3   244          LD      HL,(@LOC)
A170 B7         245          OR      A
A171 ED 52      246          SBC     HL,DE
A173 20 18      247          JR      NZ,YOUPR
A175            248          ;
A175 11 EE A3   249          LD      DE,YLOC
A178 3E 5C      250          LD      A,"¥"
A17A CD B0 A0   251          CALL    CHRPR
A17D            252          ;
A17D 3A 95 A1   253          LD      A,(ORDER)
A180 3C         254          INC     A
A181 3C         255          INC     A
A182 32 95 A1   256          LD      (ORDER),A
A185 FE 10      257          CP      16
A187 C0         258          RET     NZ
A188 AF         259          XOR     A
A189 32 95 A1   260          LD      (ORDER),A
A18C C9         261          RET
A18D            262 YOUPR
A18D 11 EE A3   263          LD      DE,YLOC
A190 3E 59      264          LD      A,"Y"
A192 C3 B0 A0   265          JP      CHRPR
A195            266 ORDER
A195 00         267          DS      1
A196            268 YOUWALK
A196 15         269          DB      $15     ; 0001_0101b
A197            270          ;
A197            271 HIKARU
A197 00 00 00   272          DS      3       ; DUMMY
A19A            273          ;
A19A 2A F0 A3   274          LD      HL,(HLOC)
A19D CD CD A0   275          CALL    CHRDEL
A1A0            276          ;
A1A0 3A 01 A2   277          LD      A,(HIKWALK)
A1A3 07         278          RLCA
A1A4 32 01 A2   279          LD      (HIKWALK),A
A1A7 E6 07      280          AND     7
A1A9 28 26      281          JR      Z,HIKPR ; YASUMI
A1AB E6 03      282          AND     3
A1AD 28 2A      283          JR      Z,HIKRND
A1AF            284          ;
A1AF 3A EE A3   285          LD      A,(YLOC)
A1B2 6F         286          LD      L,A
A1B3 3A F0 A3   287          LD      A,(HLOC)
A1B6 BD         288          CP      L
A1B7 38 03      289          JR      C,HIKARU2
A1B9 3D         290          DEC     A
A1BA 18 01      291          JR      HIKARU2+1
A1BC            292 HIKARU2
A1BC 3C         293          INC     A
A1BD 32 EA A3   294          LD      (@LOC),A
A1C0            295          ;
A1C0 3A EF A3   296          LD      A,(YLOC+1)
A1C3 6F         297          LD      L,A
A1C4 3A F1 A3   298          LD      A,(HLOC+1)
A1C7 BD         299          CP      L
A1C8            300 ;        JR      Z,HIKPR ; OCHITSUKI MODE
A1C8 38 03      301          JR      C,HIKARU3
A1CA 3D         302          DEC     A
A1CB 18 01      303          JR      HIKARU3+1
A1CD            304 HIKARU3
A1CD 3C         305          INC     A
A1CE 32 EB A3   306          LD      (@LOC+1),A
A1D1            307 HIKPR
A1D1 11 F0 A3   308          LD      DE,HLOC
A1D4 3E 48      309          LD      A,"H"
A1D6 C3 B0 A0   310          JP      CHRPR
A1D9            311 HIKRND
A1D9 CD D6 A0   312          CALL    RND
A1DC E6 01      313          AND     1
A1DE 47         314          LD      B,A
A1DF EE 01      315          XOR     1
A1E1 4F         316          LD      C,A
A1E2            317          ;
A1E2 3A F0 A3   318          LD      A,(HLOC)
A1E5 80         319          ADD     A,B
A1E6 91         320          SUB     C
A1E7 FE 27      321          CP      39
A1E9 30 03      322          JR      NC,HIKRND2
A1EB 32 EA A3   323          LD      (@LOC),A
A1EE            324 HIKRND2
A1EE CD D6 A0   325          CALL    RND
A1F1 E6 01      326          AND     1
A1F3 47         327          LD      B,A
```

```
A1F4 3A F1 A3   328          LD      A,(HLOC+1)
A1F7 80         329          ADD     A,B
A1F8 FE 17      330          CP      23
A1FA 30 D5      331          JR      NC,HIKPR
A1FC 32 EB A3   332          LD      (@LOC+1),A
A1FF 18 D0      333          JR      HIKPR
A201            334 HIKWALK
A201 73         335          DB      $73     ; 0111_0011b
A202            336          ;
A202            337 MEARY
A202 00 00 00   338          DS      3       ; DUMMY
A205            339          ;
A205 2A F2 A3   340          LD      HL,(MELOC)
A208 CD CD A0   341          CALL    CHRDEL
A20B            342          ;
A20B 3A 46 A2   343          LD      A,(MEAWALK)
A20E 07         344          RLCA
A20F 32 46 A2   345          LD      (MEAWALK),A
A212 E6 01      346          AND     1
A214 28 15      347          JR      Z,MEARYY
A216            348 MEARYX
A216 3A F4 A3   349          LD      A,(ZLOC)
A219 6F         350          LD      L,A
A21A 3A F2 A3   351          LD      A,(MELOC)
A21D BD         352          CP      L
A21E 28 0B      353          JR      Z,MEARYY
A220 30 03      354          JR      NC,MEARY2
A222 3C         355          INC     A
A223 18 01      356          JR      MEARY2+1
A225            357 MEARY2
A225 3D         358          DEC     A
A226 32 EA A3   359          LD      (@LOC),A
A229 18 13      360          JR      MEAPR
A22B            361          ;
A22B            362 MEARYY
A22B 3A F5 A3   363          LD      A,(ZLOC+1)
A22E 6F         364          LD      L,A
A22F 3A F3 A3   365          LD      A,(MELOC+1)
A232 BD         366          CP      L
A233 28 E1      367          JR      Z,MEARYX
A235 30 03      368          JR      NC,MEARY3
A237 3C         369          INC     A
A238 18 01      370          JR      MEARY3+1
A23A            371 MEARY3
A23A 3D         372          DEC     A
A23B 32 EB A3   373          LD      (@LOC+1),A
A23E            374 MEAPR
A23E 11 F2 A3   375          LD      DE,MELOC
A241 3E 4D      376          LD      A,"M"
A243 C3 B0 A0   377          JP      CHRPR
A246            378 MEAWALK
A246 AA         379          DB      $AA     ; 1010_1010b
A247            380          ;
A247            381 ZENJI
A247 00 00 00   382          DS      3       ; DUMMY
A24A            383          ;
A24A 3A 7D A2   384          LD      A,(ZSLEEP)
A24D B7         385          OR      A
A24E 28 05      386          JR      Z,ZENJI2
A250 3D         387          DEC     A
A251 32 7D A2   388          LD      (ZSLEEP),A
A254 C9         389          RET
A255            390 ZENJI2
A255 2A F4 A3   391          LD      HL,(ZLOC)
A258 CD CD A0   392          CALL    CHRDEL
A25B            393          ;
A25B CD D6 A0   394          CALL    RND     ; SLEEP?
A25E FE E8      395          CP      $E8
A260 38 0D      396          JR      C,ZENJI3
A262 E6 1F      397          AND     31      ; MAX_SLEEP 31 COUNTS
A264 32 7D A2   398          LD      (ZSLEEP),A
A267 CD 1E 20   399          CALL    #LOC
A26A 3E 7A      400          LD      A,"z"
A26C C3 F4 1F   401          JP      #PRINT
A26F            402 ZENJI3
A26F 21 F4 A3   403          LD      HL,ZLOC
A272 CD E3 A0   404          CALL    RNDXY
A275            405 ZENPR
A275 11 F4 A3   406          LD      DE,ZLOC
A278 3E 5A      407          LD      A,"Z"
A27A C3 B0 A0   408          JP      CHRPR
A27D            409 ZSLEEP
A27D 00         410          DS      1
A27E            411          ;
A27E            412 CHOUROU
A27E 00 00 00   413          DS      3       ; DUMMY
A281            414          ;
A281 3A B4 A2   415          LD      A,(CSLEEP)
A284 B7         416          OR      A
A285 28 05      417          JR      Z,CHOUROU2
A287 3D         418          DEC     A
A288 32 B4 A2   419          LD      (CSLEEP),A
A28B C9         420          RET
A28C            421 CHOUROU2
A28C 2A F6 A3   422          LD      HL,(CLOC)
A28F CD CD A0   423          CALL    CHRDEL
A292            424          ;
A292 CD D6 A0   425          CALL    RND     ; SLEEP?
A295 FE D0      426          CP      $D0
A297 38 0D      427          JR      C,CHOUROU3
A299 E6 07      428          AND     7       ; MAX_SLEEP 7 COUNTS
A29B 32 B4 A2   429          LD      (CSLEEP),A
A29E CD 1E 20   430          CALL    #LOC
A2A1 3E 63      431          LD      A,"c"
A2A3 C3 F4 1F   432          JP      #PRINT
A2A6            433 CHOUROU3
A2A6 21 F6 A3   434          LD      HL,CLOC
A2A9 CD E3 A0   435          CALL    RNDXY
A2AC            436 CHOPR
A2AC 11 F6 A3   437          LD      DE,CLOC
A2AF 3E 43      438          LD      A,"C"
A2B1 C3 B0 A0   439          JP      CHRPR
A2B4            440 CSLEEP
A2B4 00         441          DS      1
A2B5            442          ;
A2B5            443 JUNJI
A2B5 00 00 00   444          DS      3       ; DUMMY
A2B8            445          ;
A2B8 2A FC A3   446          LD      HL,(JLOC)
```

```
A2BB CD CD A0   447          CALL    CHRDEL
A2BE            448          ;
A2BE 3A 19 A3   449          LD      A,(JWALK)
A2C1 07         450          RLCA
A2C2 32 19 A3   451          LD      (JWALK),A
A2C5 E6 01      452          AND     1
A2C7 28 15      453          JR      Z,JUNY
A2C9            454 JUNX     ;
A2C9 21 EC A3   455          LD      HL,ULOC
A2CC CD F9 A2   456          CALL    MEDIUM  ; X MEDIUM
A2CF 3A FC A3   457          LD      A,(JLOC)
A2D2 BD         458          CP      L
A2D3 38 03      459          JR      C,JUNJI2
A2D5 3D         460          DEC     A
A2D6 18 01      461          JR      JUNJI2+1
A2D8            462 JUNJI2
A2D8 3C         463          INC     A
A2D9 32 EA A3   464          LD      (@LOC),A
A2DC 18 13      465          JR      JUNPR
A2DE            466 JUNY     ;
A2DE 21 ED A3   467          LD      HL,ULOC+1
A2E1 CD F9 A2   468          CALL    MEDIUM  ; Y MEDIUM
A2E4 3A FD A3   469          LD      A,(JLOC+1)
A2E7 BD         470          CP      L
A2E8 38 03      471          JR      C,JUNJI3
A2EA 3D         472          DEC     A
A2EB 18 01      473          JR      JUNJI3+1
A2ED            474 JUNJI3
A2ED 3C         475          INC     A
A2EE 32 EB A3   476          LD      (@LOC+1),A
A2F1            477 JUNPR
A2F1 11 FC A3   478          LD      DE,JLOC
A2F4 3E 4A      479          LD      A,"J"
A2F6 C3 B0 A0   480          JP      CHRPR
A2F9            481 MEDIUM
A2F9 06 08      482          LD      B,8
A2FB D9         483          EXX
A2FC AF         484          XOR     A
A2FD 57         485          LD      D,A
A2FE 21 00 00   486          LD      HL,$0000
A301 D9         487          EXX
A302            488 MED2
A302 7E         489          LD      A,(HL)  ;HL'
A303 D9         490          EXX
A304 5F         491          LD      E,A
A305 19         492          ADD     HL,DE
A306 D9         493          EXX
A307 23         494          INC     HL      ;HL'
A308 23         495          INC     HL      ;HL'
A309 10 F7      496          DJNZ    MED2
A30B            497          ;
A30B D9         498          EXX
A30C CB 3C      499          SRL     H       ; SRL HL
A30E CB 1D      500          RR      L       ; HL/2
A310 CB 3C      501          SRL     H
A312 CB 1D      502          RR      L       ; HL/4
A314 CB 3C      503          SRL     H
A316 CB 1D      504          RR      L       ; HL/8
A318            505          ;
A318 C9         506          RET             ; L=MEDIUM 8 CHARACTERS
A319            507 JWALK
A319 F0         508          DB      $F0     ; 1111_0000b
A31A            509          ;
A31A            510 KOMURA
A31A 00 00 00   511          DS      3       ; DUMMY
A31D            512          ;
A31D 3A 99 A3   513          LD      A,(KSLEEP)
A320 B7         514          OR      A
A321 28 05      515          JR      Z,KOMURA2
A323 3D         516          DEC     A
A324 32 99 A3   517          LD      (KSLEEP),A
A327 C9         518          RET
A328            519 KOMURA2
A328 3A 9A A3   520          LD      A,(KOMWALK)
A32B 07         521          RLCA
A32C 32 9A A3   522          LD      (KOMWALK),A
A32F E6 01      523          AND     1
A331 C8         524          RET     Z
A332            525          ;
A332 2A F8 A3   526          LD      HL,(KLOC)
A335 CD CD A0   527          CALL    CHRDEL
A338            528          ;
A338 CD D6 A0   529          CALL    RND     ; SLEEP?
A33B FE F8      530          CP      $F8
A33D 38 0F      531          JR      C,KOMURA3
A33F CB 3F      532          SRL     A       ; /2
A341 CB 3F      533          SRL     A       ; /4
A343 32 99 A3   534          LD      (KSLEEP),A
A346 CD 1E 20   535          CALL    #LOC
A349 3E 6B      536          LD      A,"k"
A34B C3 F4 1F   537          JP      #PRINT
A34E            538 KOMURA3
A34E 2A F8 A3   539          LD      HL,(KLOC)
A351 3E 1E      540          LD      A,30
A353 BD         541          CP      L       ;X=30?
A354 38 24      542          JR      C,KOMXL
A356 3E 0E      543          LD      A,14
A358 BC         544          CP      H       ;Y=14?
A359 38 31      545          JR      C,KOMYU
A35B 3E 08      546          LD      A,8
A35D BD         547          CP      L       ;X=8?
A35E 30 1D      548          JR      NC,KOMXR
A360 BC         549          CP      H       ;Y=8?
A361 30 26      550          JR      NC,KOMYD
A363            551          ;
A363 3E 09      552          LD      A,9
A365 BD         553          CP      L       ;X=9?
A366 20 03      554          JR      NZ,KOMURA4
A368 BC         555          CP      H       ;Y=9?
A369 20 19      556          JR      NZ,KOMY
A36B            557 KOMURA4
A36B 3E 1E      558          LD      A,30
A36D BD         559          CP      L       ;X=30?
A36E 20 05      560          JR      NZ,KOMX
A370 3E 0E      561          LD      A,14
A372 BC         562          CP      H       ;Y=14?
A373 20 0F      563          JR      NZ,KOMY
```

```
A375            564 KOMX
A375 3E 0C      565          LD      A,12
A377 BC         566          CP      H
A378 30 03      567          JR      NC,KOMXR
A37A            568 KOMXL
A37A 2D         569          DEC     L
A37B 18 01      570          JR      KOMXR+1
A37D            571 KOMXR
A37D 2C         572          INC     L
A37E 7D         573          LD      A,L
A37F 32 EA A3   574          LD      (@LOC),A
A382 18 0D      575          JR      KOMPR
A384            576 KOMY
A384 3E 14      577          LD      A,20
A386 BD         578          CP      L
A387 30 03      579          JR      NC,KOMYU
A389            580 KOMYD
A389 24         581          INC     H
A38A 18 01      582          JR      KOMYU+1
A38C            583 KOMYU
A38C 25         584          DEC     H
A38D 7C         585          LD      A,H
A38E 32 EB A3   586          LD      (@LOC+1),A
A391            587 KOMPR
A391 11 F8 A3   588          LD      DE,KLOC
A394 3E 4B      589          LD      A,"K"
A396 C3 B0 A0   590          JP      CHRPR
A399            591 KSLEEP
A399 00         592          DS      1
A39A            593 KOMWALK
A39A 11         594          DB      $11     ; 0001_0001b
A39B            595          ;
A39B            596 ATUSI
A39B 00 00 00   597          DS      3       ; DUMMY
A39E            598          ;
A39E 3A E6 A3   599          LD      A,(ATUWALK)
A3A1 07         600          RLCA
A3A2 32 E6 A3   601          LD      (ATUWALK),A
A3A5 E6 01      602          AND     1
A3A7 C8         603          RET     Z
A3A8            604          ;
A3A8 2A FA A3   605          LD      HL,(ALOC)
A3AB CD CD A0   606          CALL    CHRDEL
A3AE            607          ;
A3AE AF         608          XOR     A
A3AF BD         609          CP      L       ;X=0?
A3B0 20 07      610          JR      NZ,ATUSI2
A3B2 3E 17      611          LD      A,23
A3B4 BC         612          CP      H       ;Y=23?
A3B5 28 0B      613          JR      Z,ATUX
A3B7 18 18      614          JR      ATUY
A3B9            615 ATUSI2
A3B9 3E 27      616          LD      A,39
A3BB BD         617          CP      L       ;X=39?
A3BC 20 04      618          JR      NZ,ATUX
A3BE AF         619          XOR     A
A3BF BC         620          CP      H       ;Y=0?
A3C0 20 0F      621          JR      NZ,ATUY
A3C2            622 ATUX
A3C2 3E 0C      623          LD      A,12
A3C4 BC         624          CP      H
A3C5 38 03      625          JR      C,ATUXR
A3C7            626 ATUXL
A3C7 2D         627          DEC     L
A3C8 18 01      628          JR      ATUXR+1
A3CA            629 ATUXR
A3CA 2C         630          INC     L
A3CB 7D         631          LD      A,L
A3CC 32 EA A3   632          LD      (@LOC),A
A3CF 18 0D      633          JR      ATUPR
A3D1            634 ATUY
A3D1 3E 14      635          LD      A,20
A3D3 BD         636          CP      L
A3D4 38 03      637          JR      C,ATUYU
A3D6            638 ATUYD
A3D6 24         639          INC     H
A3D7 18 01      640          JR      ATUYU+1
A3D9            641 ATUYU
A3D9 25         642          DEC     H
A3DA 7C         643          LD      A,H
A3DB 32 EB A3   644          LD      (@LOC+1),A
A3DE            645 ATUPR
A3DE 11 FA A3   646          LD      DE,ALOC
A3E1 3E 41      647          LD      A,"A"
A3E3 C3 B0 A0   648          JP      CHRPR
A3E6            649 ATUWALK
A3E6 55         650          DB      $55     ; 0101_0101b
A3E7            651          ;
A3E7            652 MASTER
A3E7 00         653          DS      1
A3E8 C9         654          RET
A3E9            655          ;
A3E9            656 @WIDTH
A3E9 00         657          DS      1
A3EA            658 @LOC
A3EA 00 00      659          DS      2
A3EC            660 ULOC
A3EC 00 00      661          DW      $0000
A3EE            662 YLOC
A3EE 05 12      663          DW      $1205
A3F0            664 HLOC
A3F0 12 05      665          DW      $0512
A3F2            666 MELOC
A3F2 03 08      667          DW      $0803
A3F4            668 ZLOC
A3F4 10 20      669          DW      $2010
A3F6            670 CLOC
A3F6 10 16      671          DW      $1610
A3F8            672 KLOC
A3F8 20 0F      673          DW      $0F20
A3FA            674 ALOC
A3FA 10 10      675          DW      $1010
A3FC            676 JLOC
A3FC 05 09      677          DW      $0905
A3FE            678 MALOC
A3FE 00 00      679          DW      $0000
```

★(で)のショートプロぱーてぃ その28

# ヴィジュアルである

Komura Satoshi 古村　聡

カラオケは好きだけれど，音楽は苦手という（で）氏。そこへ届いた投稿は，「指板指南」というギターのコード記憶練習プログラム（?）。苦手なものでもなんでもござれ，「珍客万来」大歓迎のショートプロぱーてぃであります。

illustration : T.Takahashi

あけましてー，おめでとうございまーすっ。今年もどうぞよろしくお願いします，っと。

さて，1992年，パソコンはヴィジュアルであるっ！　……といまさら力コブ作っていうのが恥ずかしいくらい，いまのパソコンソフトってえのはヴィジュアル方面へと走っているのであります。特に，今月の特集でもあるウィンドウ環境なんてぇのは，もうまっさかり。

C>WHERE /a PARTY01
C:¥ARTIC
C>COPY ¥ARTIC¥PARTY01 A:¥

なんてコマンド君でハナモゲラしていたのが，いまではもう検索だろうがファイルコピーだろうがマウスひとつですーいすい。アイコンやらメニューやらをクリッククリックですんでしまうんだから，楽チンたらありゃしませんわな。おそらくは，これからしばらくウィンドウ環境全盛の時代となることでしょう，と私ならずとも皆さん思っておられることでしょう。

ところがどっこい，これは街で聞いた話なんだけど。いま，時代はウィンドウの時代。が，未来はそうではないんだそうな。

理由）

ウィンドウ環境では，なにかパソコンにやらせるときには与えられたメニューのなかから選ぶわけだな。ところがどっこい，音声認識装置が実用化されてしまうと……。

コマンド入力の場合。

「お～い，今月のショートプロの原稿，見つけてフロッピーに入れておいてねー」

「ほいほい了解」

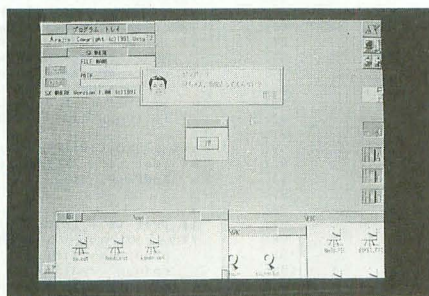
メニュー形式の場合。

「いま，作りかけのプログラムの山と，ASKの辞書があって，CONFIG.SYSと……」

「……いいから，今月の原稿探して」

「えーっと，そのディレクトリには今月の原稿と先月の原稿と先々月の原稿と……」

珍客万来

指板指南

「今月の原稿だってば」

「今月の原稿をどうするんですかぁ？」

「フロッピーに転送……」

「えーっと，いまフロッピーにはHuman.SYSとぉ……」

「いいかげんにせんかぁーいっ！」

がっちゃーん（とちゃぶ台がひっくり返る）

……と，いうわけで実は“未来はふたたびコマンド入力君のもの”なんだそうな。うーむ。あんだすたん？

**リスト1　chinkyaku.c**

```
 1: /**************************************************
 2: /*   X版 きょーふのすいよーび☆ だ!
 3: /*
 4: /*      by DAi  1991/11/06
 5: /**************************************************
 6: #include   <iocslib.h>
 7: #include   <doslib.h>
 8: #include   <basic.h>
 9: #include   <stdio.h>
10: #include   <sxlib.h>
11:
12: typedef   struct gl_val {
13:           int        eventmask;    /* イベントマスク
14:           tsevent    eventrec;     /* イベントレコード
15:           int        activflag;    /* アクティブフラグ
16:           int        taskid;       /* タスクID
17:           window     *windowptr;   /* ウィンドウレコードへのポインタ
18:           rect       wsize;        /* ウィンドウのサイズ
19: } gl_val;                          /*. . . とここはサンプルどおり
20:
21: void    nullEvent();
22: void    mouseLDownEvent(gl_val *);
23: void    updateEvent(gl_val *);
24: void    activateEvent(gl_val *);
25: void    TaskEvent(gl_val *);
26: int     Init(gl_val *);
27: void    EndPr(int, gl_val *);
28: void    msgset();
29:
30: /*文字列の形式についてはマニュアルのグラフマン(文字列)参照*/
31: char    wtitle[] = "¥008珍客万来";    /* ウィンドウタイトル    */
32: char    _sxkernelcomm[] = "sxwdb.x -D -L0";   /* カーネル起動コマンド    */
33: char    mes0[] = "ピンポーン¥n";
34: char    mes1[] = "速達でーす。ハンコお願いします。¥n";
35: char    mes2[] = "奥さん。スイカいらねぇか?¥n";
36: char    mes3[] = "あなたの幸せを祈らせてください。¥n";
37: char    mes4[] = "ダスキンでーす。¥n";
38: char    mes5[] = "兄ちゃん、新聞とってくんない?¥n";
39: char    mes6[] = "……(走って逃げた)…¥n";
40: char    mes7[] = "新聞の集金でーす。¥n";
41: int     pat;
42: char    charbuf[255];
43: int     rndnum;
44:
45: int   main()
46: {
47:    gl_val   gv;
48:    int   ret;
49:
50:    if ((ret = Init(&gv)) < 0)
51:       EndPr(ret, &gv);    /* 初期化に失敗したので終了する    */
52:    while (1) {          /* タスクマンのイベントを得る    */
53:       TSEventAvail(gv.eventmask, &gv.eventrec);
54:       switch (gv.eventrec.what) {
```

## 窓には人がやってくる

……とかなんとかいいつつ，実はっ！なんと今月の1本目のプログラムはショートプロ初のSX-WINDOW用のプログラムだったりするのです(もう，文章の脈略がぐちゃぐちゃだがね)。さてさてお立ち会い。

**珍客万来.X for SX-WINDOW**

**(要Cコンパイラ)**

**神奈川県　山下　大作**

えー，このプログラムはCコンパイラで書かれたSX-WINDOW用の冗談プログラムです。したがってこのプログラムの実行には2Mバイト以上のRAMを積んだX68000と，SX-WINDOWのシステム，C compiler PRO-68K ver.2.0，1991年1月号付録（謹賀新年PRO-68K）に収録されているsxlib.lと広い心が必要になります。

実行するためには，まずはリスト1をエディタで入力してchinkyaku.cという名前でセーブしておいてください。

次にCのライブラリのあるディレクトリ（ここでは¥libということにしておきます）にsxlib.lと__mainr.oを，インクルードファイルのあるディレクトリ（¥include）にsxdef.hを入れておきます。そして，

```
A>cc /Fc chinkyaku.c
A>lk chinkyaku.o ¥lib¥__mainr.o
  ¥lib¥clib.l  ¥lib¥sxlib.l
  ¥lib¥floatfnc.l  ¥lib¥iocslib.l
  ¥lib¥doslib.l
A>ren chinkyaku.x 珍客万来.x
```

以上で珍客万来.xの完成であります。

SX-WINDOW上でこのプログラムをクリックすると押しボタン（？）が表示されます。ボタンを押すとチャイムとともにやってくるのは誰でしょう……って，誰だよこんな原稿書きにくいプログラム書いたのは。まあ，ありがちなプログラムなんですけどね～。なんなんだよ，このプログラムはよー。え？　なに？　SX-WINDOWのエラー音を聞いてこのプログラムを思いついた？　苦情出ても知らないからねー，こんな冗談プログラム載せて。

しかしSX-WINDOWでショートプロとは無謀なことをしたもんだねぇ。スケルトン（プログラムの骨格の部分ね）だけでも200行ぐらいにはなってしまうもんね。面倒臭い決まりごとも多いし。ツールもC compiler PRO-68Kとかしかないし。その点，SX-WINDOWの最大のライバル(？)，MS-WINDOWSとかにはvisual BASICあり，TurboPASCALありで誰でも簡単にプロ

```
 55:             case E_IDLE:        nullEvent();        break;
 56:             case E_MSLDOWN:     mouseLDownEvent(&gv);    break;
 57:             case E_UPDATE:      updateEvent(&gv);   break;
 58:             case E_ACTIVATE:    activateEvent(&gv);    break;
 59:             case E_SYSTEM1:
 60:             case E_SYSTEM2:     TaskEvent(&gv);        break;
 61:         }
 62:     }
 63: }
 64:
 65: /************************************************
 66: /*    ヌルイベント
 67: /************************************************
 68: void    nullEvent()
 69: {
 70:  /*なにもしない*/
 71: }
 72:
 73: /************************************************
 74: /*    マウスレフトダウンイベント
 75: /************************************************
 76: void    mouseLDownEvent(gl_val *gp)
 77: {
 78:     int    ret;
 79:     int    reta;
 80:
 81:     /* 自分のウィンドウ上かを調べる    */
 82:     if (gp->windowptr == (window *)(gp->eventrec.whom)) {
 83:         /* マウスダウンイベントを取り除く    */
 84:         TSGetEvent(gp->eventmask, &gp->eventrec);
 85:         if (gp->activflag == 0) {
 86:             WMSelect(gp->windowptr); /* ウィンドウをアクティブに */
 87:             ret = WMFind(*(point_t *)(&gp->eventrec.whom2),
 88:                 (window **)(&reta));
 89:             if (ret != W_INDRAG)    return;    /* タイトルバー以外の時はリターン */
 90:             if (EMLStill() == 0)    return;    /* マウスが離されていたらリターン */
 91:         }
 92:         ret = SXCallWindM(gp->windowptr, &gp->eventrec);
 93:         if (ret == W_INCLOSE){        /* クローズボタンが押された */
 94:             EndPr(0, gp);        /* 終了する        */
 95:
 96:         }
 97:         if (ret == W_ININSIDE){
 98:         /*ウインドウ内で左ボタンが押された*/
 99:             randomize(TIMEGET());
100:             rndnum =rnd() * 7;
101:             strcpy(charbuf,mes0);
102:                 switch(rndnum){
103:                     case 0:strcat(charbuf,mes1);break;
104:                     case 1:strcat(charbuf,mes2);break;
105:                     case 2:strcat(charbuf,mes3);break;
106:                     case 3:strcat(charbuf,mes4);break;
107:                     case 4:strcat(charbuf,mes5);break;
108:                     case 5:strcat(charbuf,mes6);break;
109:                     case 6:strcat(charbuf,mes7);break;
110:                 }
111:                 DMError(0x0201,charbuf);
112:
113:         }
114:     }
115: }
116:
117: /************************************************
118: /*    アップデートイベント
119: /************************************************
120: void    updateEvent(gl_val *gp)
121: {
122:     int    ret;
123:                                     /* 自分のウィンドウかを調べる    */
124:     if (gp->windowptr == (window *)(gp->eventrec.whom)) {
125:         WMUpdate(gp->windowptr);        /* アップデートリージョンをセット */
126:                                     /* <graph> をカレントにセット */
127:         GMSetGraph((graph *)(gp->windowptr));
128:         /*これをしないと回りのタスクが止まるみたい*/
129:         GMFontMode(0);
130:         msgset();
131:         WMUpdtOver(gp->windowptr);    /* リージョンを戻す    */
132:     }
133: }
134:
135: /************************************************
136: /*    アクティベートイベント
137: /************************************************
138: void    activateEvent(gl_val *gp)
139: {
140:     if (gp->windowptr == (window *)(gp->eventrec.whom))
141:     gp->activflag = 1;
142:     else    gp->activflag = 0;
143: }
144:
145: /************************************************
146: /*    タスクマンのイベント
147: /************************************************
148:
149: void    TaskEvent(gl_val *gp)
150: {
151:     int    sendtask;
152:     int    ret, i;
153:
154:     switch (gp->eventrec.what2) {
155:       case CLOSEALL:            /* 全ウィンドウのクローズ */
156:       case ENDTSK:               /* 全タスクの終了    */
157:         EndPr(0, gp);
158:         break;
159:       case WINDOWSELECT:            /* ウィンドウのセレクト    */
160:         WMSelect(gp->windowptr);
161:         break;
162:     }
163: }
164:
165: /************************************************
166: /*        初期処理
167: /************************************************
168: int    Init(gl_val *gp)
169: {
170:     task    tbuff;
171:     int    ret;
172:
173:     gp->activflag = 0;gp->windowptr = 0;gp->eventmask = 0xffff;
```

グラム組めるし，ショートプロだってできるんだよね。はやくSX-WINDOWにもいろんなツールが出て，SX-WINDOWでもショートプロが作れるようになるといいんだけどね。

```
174:     TSSetAbort(&EndPr, gp);               /* アボート終了処理を登録 */
175:     gp->taskid  = TSGetID();              /* タスクIDを得る    */
176:     TSGetTdb(&tbuff, -1);                 /* タスク管理テーブルを得る */
177:     ret = TSTakeParam(&tbuff.command, &gp->wsize, 0, 0, 0, 0);
178:     if ((ret & 1) == 0) {
179:        *(long *)(&gp->wsize.left) = TSGetWindowPos();/*ウィンドウの開始位置をget*/
180:        gp->wsize.right = gp->wsize.left + 0x60;/*横長さ*/
181:        gp->wsize.bottom = gp->wsize.top + 0x40;/*縦長さ*/
182:
183:     }
184:     pat = 0;
185:     gp->windowptr = (window *)WMOpen((window *)(0), &gp->wsize,
186:        (LASCII *)(&wtitle), 0, 32<<4, (window *)(-1), -1, gp->taskid);
187:        /*ウインドウレコードへのポインタを得る*/
188:     if (gp->windowptr == 0)   return(-1);
189:        /*getに失敗したらエラーを返す*/
190:     GMSetGraph((graph *)(gp->windowptr));
191:        /*グラフマンにウインドウへのレコードポインタを渡して*/
192:     GMFontMode(0);         /*描画モードはPSET*/
193:     WMShow(gp->windowptr);    /*ウインドウを描画*/
194:              /*(ウインドウを可視属性にセット)*/
195:     GMSetGraph((graph *)(gp->windowptr));
196:     msgset();
197:     return(0);
198: }
199:
200: void   msgset()
201: {
202:     GMMove(20<<16 | 1*12);   GMDrawStrZ("┌───┐");
203:     GMMove(20<<16 | 2*12);   GMDrawStrZ("│ 押 │");
204:     GMMove(20<<16 | 3*12);   GMDrawStrZ("└───┘");
205: }
206:
207: /***********************************************
208: /*    終了処理
209: /***********************************************
210: void   EndPr(int code, gl_val *gp)
211: {
212:     if (gp->windowptr != 0) {
213:        WMDispose(gp->windowptr);    /* ウィンドウを廃棄する    */
214:     }
215:     exit(code);
216: }
217:
218:
```

## 指板指南でぺろんぽろん

続いて今月の2本目にまいりましょう。2本目のプログラムはX-BASIC用のギターの友の友プログラムで「指板指南」です。

**SHIBAN.BAS for X68000**
**(X-BASIC)**
**神奈川県 野田 敏之**

ギターやベースでは，何弦の何フレットが何の音かを暗記していれば，上達の近道になります，んだそうなのです。このプログラムは指板上の音（12フレットまで）を暗記するためのプログラムなのです。

プログラムを実行すると，ギターかベースかを聞いてきます。ギターなら0，ベースなら1を入力します。入力すると画面上に指板のモデルが現れ“●”で示す音を聞いてきますのでその音をC～B（#は＋，bは−）で入力してください。たとえば，“●”がギターの1弦2フレットにあれば，その音はF#(Gb)ですから半角で“F＋”か“G−”と入力すればいいわけです。

正解したときには“正解”の文字が，間違いのときには正しい答えが表示されます。

これがエンドレスで続きます。止めたいときには“Q”を入力してください。

90行のチューニングデータと140行のSCの値（弦の数）を変えれば「変則チューニング」や「6弦ベース」,「マンドリン」や「大正琴」用にもできるはず，です。

……わからんよーん。ここまでイッキに説明を書いて（しかもほとんど投稿原稿そのまんま）しまったのはほかでもない。このテの話題にはじぇんじぇんついていけないからなんだよーん。だってギターなんてやったことないし，学校の音楽の成績はへろへろのぽよぽよのぺもぺもだったしー。第一，学校の音楽の授業でマンドリンやったけど，いちばん細い弦を切って「“必殺仕事人っ！”きゅきゅきゅきゅきゅっ！ ぺよーん（……わ，わかるかなぁ？）」などと遊んでしまって，グランド20周させられた思い出があるだけで，もうなんにも覚えてないんだよーん。ぺよよ～ん。

もっとマトモにやれって？ じゃあ,「東京ラテン系セニョリータ」をカラオケで歌って，3日3晩ノドがつぶれてた話でもするか。それとも「がんばれタカハシ」か？

とにかく役に立つプログラムなんだと思います。たぶん。ギター，ベース関係の方は使ってみてください。はい。

ショートプロはどんなジャンルのプログラムでも受け付けます。燃えよキーボード！ 来たれ，熱き作品よ。また来月。

**リスト2 SHIBAN.BAS**

```
10 /*          指板指南  for X68000 X-BASIC   1991,9,8
20 /*          Programmed by Toshiyuki "material" Noda
30 /*
40 screen 1,1,1,1:randomize(val(right$(time$,2))*100)
50 str sha(11)={"C","C+","D","D+","E","F", /* #音階データ
60              "F+","G","G+","A","A+","B"}
70 str fla(11)={"C","D-","D","E-","E","F", /* ♭音階データ
80              "G-","G","A-","A","B-","B"}
90 str tug(5)={"E","B","G","D","A","E"}    /* ﾁｭｰﾆﾝｸﾞ ﾃﾞｰﾀ
100 str ans:char sc,lp
110 /*
120 repeat
130   input "GUITAR=0 / BASS=1";sc
140   sc=-6*(sc=0)-4*(sc=1):cls               /* ｷﾞﾀｰ6弦 ﾍﾞｰｽ4弦
150 until sc
160 /*
170 if sc=4 then {                     /* ｷﾞﾀｰ→ﾍﾞｰｽ ﾁｭｰﾆﾝｸﾞ変換
180   for i=0 to 3:tug(i)=tug(i+2):next
190 }
200 /*
210 print "┌┬";:for i=0 to 11:print "─┬";:next:print "─
220 for j=0 to sc-3
230 print "│├";:for i=0 to 11:print "─┼";:next:print "─
240 next
250 print "└┴";:for i=0 to 11:print "─┴";:next:print "─
260 print spc(6);:for i=0 to 3:print spc(6);"・";:next
270 print spc(10);":":print spc(22);
280 if sc=6 then print "Guitar" else print "  Bass
290 print " ":print spc(25);"Q=Quit":console 10,21,0
300 /*
310 repeat
320   sr=int(rnd()*sc):fr=int(rnd()*13)
330   symbol(fr*32,sr*16,"●",1,1,1,15,0)
340   print sr+1;"弦";fr;"フレット";
350   input ans:ans=strupr(ans)
360   if ans="Q" then lp=1 else {
370     if note(0)=ans then print "正解" else {
380       if note(1)=ans then print "正解" else {
390         print "間違い！答えは ";:ans=note(0)
400         if note(1)=ans then print ans else {
410           print ans;" (";note(1);")
420         }
430       }
440     }
450   }
460   symbol(fr*32,sr*16,"●",1,1,1,0 ,0):print " "
470 until lp
480 screen 1,1,1,1:end
490 /*
500 func str note(fg):str sca(11)
510   if fg then {
520     for i=0 to 11:sca(i)=fla(i):next    /* fg=1  ♭
530   } else {
540     for i=0 to 11:sca(i)=sha(i):next    /* fg<>1 #
550   }
560   for i=0 to 11
570     if sca(i)=tug(sr) then {
580       j=i:j=j+fr:while j>11:j=j-12:endwhile
590     }
600   next
610 return(sca(j))
```

# (で)のぱーてぃハンズ

## キー入力の傾向と対策

今月はいままでとはちと方針を変えて，ゲーム作りに必要な基礎知識を扱ってみましょう。

X68000からパソコンを始めた人，あるいは他機種から移ってきた人，とにかく初めてX68000に触った人がゲームを作ろうと思った。で，X-BASICを使おうとするといちばん悩むのがキー入力でありましょう。知らないとなかなかうまくないんですね，これが。

まず，だいたい最初にマニュアルに見つける，あるいは他機種から来た人が最初に考えるのがinkey$。なぜかというと他機種では，

a$＝inkey$

ってやると，“リアルタイムキー入力”になるのがほとんどだからなのです。ところがどっこい，なぜかX68000では，

「inkey$はキーボードに文字が入るまで待つ」キー入力であったりするのですね。これは他機種でいう，

a$＝input$(1)

などにあたります。であるからして，当然のようにinkey$を使ってしまうと，「むむーん。動かん……」となってしまうわけです，これが。さて，それではどーしたもんか。

「素直にキーボードはあきらめて，ジョイスティックやマウスを使う」

パラリ，パラリとマニュアルを見ていって，リアルタイム入力ができそうなのは……，いくつかありますね。えっと，stick( )とか，strig( )とか。あとはmsstat( )とかですね。

stickはジョイスティックの状態を，strigはジョイスティックのトリガを，msstatはマウスのボタンの状態をそれぞれ見る関数です。

X68000ユーザーならマウスは持ってるし，同時入力だってできるしー，ジョイスティックを2本つなげば対戦ゲームだってできるしー。と，“しー”調なあなたにはうってつけの方法といえるんじゃないかと思います。

しかしながら，であります。

もしかしたらジョイスティックを持っていない人だっているかもしれないのだな。どーしてもキーボードでぐりぐりやりたいんだな。キーボードじゃなくちゃヤなんだな，というわがままな要求に応えなくちゃいかんこともあるかもしれないわけです。えーい，そんな要求のめるかーい，がっちゃーん，とチャブ台をひっくり返す，というテもなくはないですが，相手が仕事の上司でどうしても断れないとか，娘を借金のカタにとられている，なんていう場合はそうもいきますまい。なんとかしてX-BASICでリアルタイムキー入力をやるしかないでしょう。

さて，それではどうしましょ。

7月号の付録「X-BASICリファレンスブック」を見た人は知っている。見てない人は知らない(かもしれない)。じっ，つっ，はっ。X-BASICのマニュアルには載っていないのですが，リアルタイムキー入力はできる，しかも2通りの方法でできてしまうのです。なんで公開しなかったんでしょうねー？　不思議。

## inkey$(0)を使う

inkey$(0)

こいつがそのキーボードから入力された文字(str型)をリアルタイムで返す未公開関数なのです。つまり，リアルタイム入力をさせるにはinkey$の代わりに，

a$＝inkey$(0)

と置き換えれば，それでOKなのですね。なーんだ。

で，2つのうちのもうひとつの方法。

未公開命令にはもうひとつキーボード関係の命令があります。それは，

keysns( )

という命令です。こいつも「X-BASICリファレンスブック」の未定義命令のページに載っています。keysns( )命令というのはキーバッファの中身を調べて，キーボードが押されたかどうかを調べる命令です。こいつでキーボードの入力があったかどうか調べて，あったらinkey$する。つまり，

if keysns( ) then a$=inkey$

で，

a$＝inkey$(0)

と同じくリアルタイムキー入力ができるのです。よかった，よかった。

ちなみにこのkeysns( )命令，image.fncの中に入っている関数ですので，basic.cnfの，

FUNC ＝ IMAGE

を“メモリが足りなくて削っちゃったよーん”などという人は，ちゃんと戻しておいてくださいね。

しかもこの未定義命令，inkey$(0)もkeysns( )も両方ともちゃーんとコンパイルもできます。なんだ，なんの問題もないですね。めでたしめでたし……，といきたいけど，あとひとつ問題があるのです。このinkey$(0)には弱点があるのです。それは，

**キー入力が溜まってしまう**

ことなのです。たとえば，inkey$(0)する前にグラフィックを描いて時間がかかったりすると，その間に押されたキーの内容がinkey$(0)するまで残ってしまうのです。さて，どうしたものでしょう。

## 空回りせよ!

溜まったキーをクリアするにはどうするのか？　XIBASICでは，

key0,""

とやればキーバッファをクリアして，溜まっていたキーの内容を消すことができたのですが……。さーて，さてさてお立ち会い。ここで取り出したるはkeysns( )命令。まずはリスト3をご覧あれ。

これは何をしているのか。まず，

for i=0 to 10000 : next

なにもせんとぐるぐる回っている。つまり時間をつぶしているわけですね。

続いて，さっき話したようにkeysns( )でキーバッファを見て，キーの内容を表示。キーバッファが空だったら“＊”を表示している。

じゃ，実行してみましょう。しばらくなにも起こりませんので，だーっとキーを入力してみてください。すると，しばらくして入力した文字が表示されてから，＊＊＊＊5＊＊＊＊とかいうふうに，＊の間にキーボードで押している文字が表示されるようになります。そうです。

しばらく待ってキーの内容が溜まってから，押されたキーを表示しているのです。

さて，ここで画面をじーっと，見る。見る。見る。見る。見る(ワープロは楽だな)。なにか気がつきませんか？　最初に溜まった文字が続けて出てくる。で，溜まったのを表示したあと，＊の間に押されてるキーがときどきポコポコと出てくる。

そうなんです。溜まっていた文字は最初に“まとめて出てくる”。ってぇ，ことはぁ，そうです，溜まっている文字をinkey$(0)が読んでいるときには，次を読んでも，keysns( )は“まだキーバッファにキーがあるよ”と教えてくれるはずなのです。そして，キーバッファのクリアにさっきのkeysns( )が使えるのです。keysns( )でキーバッファが空になるまでキー入力を取っては捨て，取っては捨てという動作を繰り返せばいいのです。

while(keysns( )):a$＝inkey$(0) : endwhile

これで，このあとa$をなんにも使わないとか，

a$＝""

としてしまえば，バッファのクリアになるのです。

キーが溜まっては困るところにこれを書いておけば，溜まっていたキーもすっと消えてくれるわけです(リスト4)。

ちなみにこれ，ちょっと工夫するとinkey$(0)だけでもできます。

## もっといろいろしたいね

さてさて，とりあえず，これでうまくキー入力ができましたね。

あとは，これで同時に“4”と“x”のキーを押したのがわかればなあ，などと考えている人もいることでしょう。市販のゲームじゃやってるもんね。キーを押しっぱなしにしたときにもっと早くキーの内容がわかればなあとか(さっきのリスト2も，どんなに早くキーを押してもキーが溜まってないと間に＊がはさまちゃったでしょ。押しっぱなしでも途中で押されてないことになっちゃうんだよね)。

そこまでいくと，はっきりいってアセンブラで新しい関数を作ってしまうしかないんですね。残念ながら。自分で作って，ショートプロにでも投稿してくださいな。

さて，来月は何の話をしましょうか？　うーむ，今月号を書いたばかりではネタが思い浮かばないな。ま，来月は来月の風邪をひくとしよう。へくし。

**リスト3**

```
1000 int i
1010 str kb
1020 for i=0 to 10000:next
1030 for i=0 to 2000
1040 if keysns() then kb=inkey$:print kb; else print"*";
1050 next
```

**リスト4**

```
1000 int i
1010 str kb
1020 for i=0 to 10000:next
1030 while(keysns()):kb=inkey$(0):endwhile
1040 for i=0 to 2000
1050 if keysns() then kb=inkey$:print kb; else print"*";
1060 next
```

X68000CARDDRV用カードゲーム

# サバイバル・ゲーム

Iketani Masahiko 池谷 昌彦

**常に勝ち続けて他人を蹴落としてでも生き残れ。そんなスリルを味わえるのがこの「サバイバル・ゲーム」。勝利を目指して切り札をうまく使いひたすら戦い続ける。負けるとわかっているときでもあきらめてはいけない。知恵と忍耐で勝負だ。**

## ゲームの生い立ち

先日，本屋をぶらついていましたら新しいトランプゲームの本を見つけました。さっそく購入して家に帰り，ページをめくっているうちに面白そうなゲームがありましたので，あっという間に作り上げてしまいました。

目に留まったゲームは，題して「生き残るのは誰か？　サバイバル・ゲーム」という名前負けしそうな作品です。実は，「LAST IN」なるゲームを作ったのですが，出来上がって遊んでみるといっこうに面白くない。なかば捨てかけていたところ，このゲームが見つかったので飛びつきました。

## 入力方法

いつものようにCARDDRVを組み込んでから，CARD2.FNC（CARD.FNCでもOK）を登録したX-BASICを起動して，リスト1を入力してください。今回のリストは，結構大きめなので気合いを入れて打ち込みましょう。デバッグも大変ですが，ゲームの面白さは保証します。

それから，このプログラムはこのままコンパイルしてもかまいません。別にインタプリタ上で動かしても，まったく速度的に支障はないと思います。好みにあわせて各自行ってください。

## 遊び方

このゲームでは，まず，4人にそれぞれ9枚のカードが配られます。カードの強さは，A(最高)，K，Q，J，……4，3，2(最低）の順です。そして，自分の最後に配られたカードのスーツが切り札となります。

ゲームの進め方は，各自1枚ずつ最初に出されたカードと同じスーツのカードを出していきます。同じスーツがない場合には，切り札のカードかそのほかのカードを出さなくてはなりません。そして，全員が場にカードを出し終わったときに，最強の切り札のカードまたは同じスーツの最強のカードを出した人が，ストックから1枚手札に加えることができ，さらに次のゲームでは最初にカードを出す権利を与えられることになります。

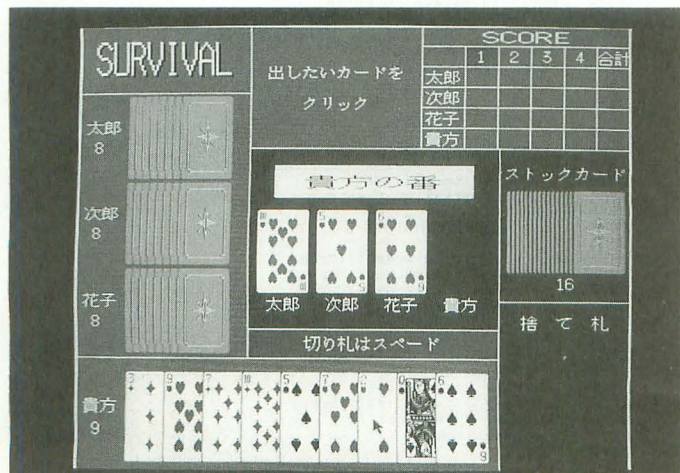
そうして，手札がなくなってしまった人は負け，最後までカードを持っている人が勝ちとなり，手持ちのカードの枚数×100点がスコアに加算されます。全部で4回勝負してスコアがいちばん多い人が勝ちになります。

## 勝負師を目指せ

ということでルールの説明をしてきましたが，やはりゲームといえども勝負には勝ちたいものですね。そこで，勝つポイントといえないまでも私がプレイするときにはこういったことをする，というようなことを少々書いてみます。

1.ゲームが開始されたらとりあえず2，3回はリードを取る（手札によってはしばらく待つ）
2.そして切り札のカードで比較的弱いカードを出して，相手に切り札のカードを出させる
3.弱いカードでしばらく様子を見ていく

ぐらいなものでしょうか。それにしても切り札のカードを返されたときの悔しさはとても言葉では表せません。また，貧弱な手持ちのカードで勝ったときの快感もなかなか気持ちがいいものです。すべてのゲームを勝ち続けるのはちょっと大変ですが，どたんばの逆転勝利もありえます。途中で投げ出さず根気よく勝負していきましょう。

## リスト1

```
  10 /*
  20 /*  SURVIVAL (要 CARD.FNC)
  30 /*        programed by M.I.,Aug.21'90
  40 /*
  50 screen 1,1,1,1:console ,,0
  60 int jj,b1,b2,b3,b4,bb,kiri,ms,f=0,rd=1
  70 dim int cc(51),c(3,8),pp(51),p(3,8),stc(15),stp(15)
  80 dim int b(3),sute(3),m(3),kei(3),ten(4,3)
  90 dim str nam(3)={ "太郎","次郎","花子","貴方" }
 100 dim str suit(3)={"スペード"," ハート"," ダイア"," クラブ"}
 110 palet(1,0)
 120 /* main program
 130 while f=0
 140            scrn()
 150            play()
 160            jd3()
 170 endwhile
 180 owari()
 190 end
 200 /* 画面作成
 210 func scrn()
 220    int i
 230    apage(3):vpage(15)
 240    fill(0,0,511,511,10)
 250    apage(2)
 260    box(0,0,511,511,15):box(1,1,510,510,15)
 270    line(2,80,159,80,15):line(161,144,509,144,15)
 280    line(161,352,391,352,15):line(160,2,160,383,15)
 290    line(320,2,320,143,15):line(2,384,391,384,15)
 300    line(392,145,392,509,15):line(393,320,509,320,15)
 310    for i=0 to 4:line(321,i*24+24,509,i*24+24,15):next
 320    for i=0 to 5:line(i*30+360,25,i*30+360,143,15):next
 330    symbol(18,19,"SURVIVAL",2,3,1,1,0)
 340    symbol(16,17,"SURVIVAL",2,3,1,15,0)
 350    symbol(376,6,"SCORE",2,1,1,15,0)
 360 for i=0 to 3:symbol(324,i*24+53,nam(i),1,1,1,15,0):next
 370 for i=1 to 4:symbol(i*30+341,29,str$(i),1,1,1,15,0):next
 380    symbol(481,29,"合計",1,1,1,15,0)
 390    for i=0 to 3:kei(i)=0:next
 400 endfunc
 410 /* play
 420 func play()
 430            while rd<5
 440                    prep()
 450                    splay()
 460                    jd2()
 470            endwhile
 480            wait(60)
 490 endfunc
 500 /*準備作業
 510 func prep()
 520    int i,j,a,b,s
 530 /* music data set
 540    if rd=1 then {
 550     m_init()
 560     for i=1 to 8:m_alloc(i,2000):m_assign(i,i):next
 570     m_trk(1,"q1@23v10o4t200l32 e")
 580     m_trk(2,"q1@23v10o4t200l32 a")
 590     m_trk(3,"q8@32v14o3t100l4  a")
 600     m_trk(4,"q5@56v15o5t 80a16e4")
 610     m_trk(5,"q7@56v14o5t 80l4  a")
 620     m_trk(6,"q8@56v15o6t 80l2  c")
 630     m_trk(7,"q6@1 v15o4t120l4db8.<c16>aaa2<ec#8.d16ggg2")
 640     m_trk(8,"q7@1 v15o4t 90l4ct60ft90l8efg2&gaaga2&ab-b-a<
d4.c>b-aefg2")
 650            }
 660 /* deal
 670 apage(1)
 680 fill((rd-1)*30+331,25,(rd-1)*30+359,47,0)
 690 fill(rd*30+331,25,rd*30+359,47,5)
 700 for i=1 to 4:symbol(i*30+341,29,str$(i),1,1,1,15,0):next
 710    if rd=1 then {
 720    symbol(184,40,"ルールの説明は",1,1,1,15,0)
 730      s=sel(176,96,1,1):if s=1 then rule()
 740    }
 750    randomize(val(mid$(time$,4,2)+right$(time$,2)))
 760    for i=0 to 51:cc(i)=i+1:next
 770    ms=16:for i=0 to 3:m(i)=9:next
 780    if s=2 or rd>1 then {
 790          er_upms()
 800          symbol(200,24,"シャッフル",1,1,1,15,0)
 810          symbol(224,56,"及び"1,1,1,15,0)
 820          symbol(200,88,"カード配布",1,1,1,15,0)
 830    }
 840    for i=0 to 99
 850          a=int(rnd()*52):b=int(rnd()*52)
 860          k=cc(a):cc(a)=cc(b):cc(b)=k
 870    next
 880    for i=0 to 51            /*Aを最高位にする
 890          if cc(i)=1  then pp(i)=13:continue
 900          if cc(i)=14 then pp(i)=26:continue
 910          if cc(i)=27 then pp(i)=39:continue
 920          if cc(i)=40 then pp(i)=52:continue
 930          pp(i)=cc(i)-1
 940    next
 950    for i=0 to 8
 960          c(0,i)=cc(i)   :p(0,i)=pp(i)
 970          c(1,i)=cc(i+9) :p(1,i)=pp(i+9)
 980          c(2,i)=cc(i+18):p(2,i)=pp(i+18)
 990          c(3,i)=cc(i+27):p(3,i)=pp(i+27)
1000    next
1010    for i=0 to 15
1020          stc(i)=cc(i+36):stp(i)=pp(i+36)
1030    next
1040    for i=0 to 51:cc(i)=0:pp(i)=0:next
1050     er_upms()
1060     if s=1 then {
1070          click()
1080            apage(0):fill(0,0,511,511,0):apage(1)
1090      }
1100      mkba()
1110 for j=0 to 2:comcd(j):next
1120 plcd():stcd()
1130 kiri=(c(3,8)-1)¥13          /* 切り札の決定
1140 symbol(212,360,"切り札は"+suit(kiri),1,1,1,15,0):m_play(6)
1150  for j=0 to 3:for i=0 to 8
1160    if (p(j,i)-1)¥13=kiri then p(j,i)=p(j,i)+100
1170  next:next
1180  for i=0 to 15
1190     if (stp(i)-1)¥13=kiri then stp(i)=stp(i)+100
1200  next
1210  for j=0 to 2:jun(j):next
1220  if rd=1 then jj=0
1230 symbol(196,174,nam(jj)+"が最初",2,1,1,1,0):m_play(5):click()
1240 endfunc
1250 /* 各トリック毎のプレー及び判定
1260 func splay()
1270   repeat
1280     b1=0:b2=0:b3=0:b4=0:bb=1
1290     ssplay()
1300     jd1()
1310     until m(0)+m(1)+m(2)=0 or m(1)+m(2)+m(3)=0 or m(2)+m(3
)+m(0)=0 or m(3)+m(0)+m(1)=0
1320 endfunc
1330 /*
1340 func ssplay()
1350   while bb<5
1360   if jj>=0 and jj<=2 then {
1370     if bb=1 then { /*bbは場へ出す順
1380          b1=first():bb=2:jj=jj+1
1390         if jj=3 then you():jj=0:bb=3:continue
1400     } else if bb=2 then {
1410          b2=com(2):bb=3:jj=jj+1
1420          if jj=3 then you():jj=0:bb=4:continue
1430     } else if bb=3 then {
1440          b3=com(3):bb=4:jj=jj+1
1450          if jj=3 then you():bb=5
1460     } else if bb=4 then b4=com(4):bb=5
1470   } else if jj=3 and bb=1 then you():jj=0:bb=2:continue
1480    endwhile
1490 endfunc
1500 /* com play as 1st player
1510 func first()
1520  int a,i,j,id,is,bc=0
1530  if m(jj)>0 then {          /*手札が無ければ飛ばす
1540    dsban(jj)
1550  if sute(jj)>0 then { /*前回勝った札があれば出してみる
1560  if sute(jj)>100 then a=(sute(jj)-101)¥13 else {
1570        a=(sute(jj)-1)¥13}
1580   for i=0 to m(jj)-1
1590       j=m(jj)-1-i
1600   if (c(jj,j)-1)¥13=a then bc=1:break
1610       next
1620   }
1630   if bc=0 then {
1640      for i=0 to m(jj)-1 /*強い切り札があれば出す
1650        if p(jj,i)-101 mod 13>8 then bc=1:break
1660      next
1670   }
1680   if bc=0 then {  /*強い札があれば出す
1690       for i=0 to m(jj)-1
1700         if (p(jj,i)-1) mod 13>7 then bc=1:break
1710       next
1720   }
1730    if bc=0 then i=int(rnd()*m(jj))
1740    is=i:m(jj)=m(jj)-1
1750    bacd(jj,is):comcd(jj)
1760    id=p(jj,is):b(jj)=id:sute(jj)=id:p(jj,is)=0:c(jj,is)=0
1770    for i=0 to 8:cc(i)=c(jj,i):pp(i)=p(jj,i):next
1780    cdleft(is)
1790    for i=0 to 8:c(jj,i)=cc(i):p(jj,i)=pp(i):next
1800    if jj=2 then wait(15) else wait(30)
1810    return(id)
1820   } else b(jj)=0:return(0)
1830 endfunc
1840 /* com play as 2nd to 4th player
1850 func com(q)
1860 int i,j,id,is,a,ap=0,sup=0
1870 if b1>100 then a=(b1-101)¥13 else if b1>0 then a=(b1-1)¥13
1880  if m(jj)>0 then {          /*手札が無ければ飛ばす
1890     dsban(jj):wait(30)
1900    for i=0 to m(jj)-1          /*最初の人と同じ札があるか
1910      if (c(jj,i)-1)¥13=a then ap=ap+1
1920    next
1930    if ap=0 then {          /*無い場合
1940      if b1>100 then {          /*切り札が出ている場合
1950      is=weakest(jj)
1960    } else {                /*切り札が出てない場合
1970     for i=0 to m(jj)-1
1980      if p(jj,i)>100 then sup=sup+1
1990     next
2000     if sup>0 then {
2010        switch q
2020        case 2:is=scom2():break
2030        case 3:is=scom3():break
2040        case 4:is=scom4()
2050        endswitch
2060     } else is=weakest(jj)
2070     }
2080     } else {          /*最初の人と同じ札がある場合
2090       switch q
2100       case 2:is=sscom2():break
2110       case 3:is=sscom3():break
2120       case 4:is=sscom4()
2130       endswitch
2140     }
```

▶ボンバーマンを買ってからというもの我が家での勉強会が絶えない。これで受験勉強は完璧と自負している次第である。　円福　貴光(18)愛知県

```
2150        m(jj)=m(jj)-1
2160        bacd(jj,is):comcd(jj)
2170        id=p(jj,is):sute(jj)=id
2180        if p(jj,is)<100 and ap=0 then id=0
2190        b(jj)=id:p(jj,is)=0:c(jj,is)=0
2200        for i=0 to 8:cc(i)=c(jj,i):pp(i)=p(jj,i):next
2210         cdleft(is)
2220         for i=0 to 8:c(jj,i)=cc(i):p(jj,i)=pp(i):next
2230          if jj=2 then wait(15) else wait(30)
2240          return (id)
2250     } else b(jj)=0:return(0)
2260 endfunc
2270 /*play you
2280 func you()
2290  int i,is,x,y,l,r,a,ap=0,bc=0
2300  if m(3)>0 then {          /*手札が無ければ飛ばす
2310 dsban(3)
2320 if bb=1 then fill(161,208,391,312,0)
2330 if b1>100 then a=(b1-101)¥13 else if b1>0 then a=(b1-1)¥13
2340   if b1>0 then {
2350    for i=0 to m(3)-1
2360      if (c(3,i)-1)¥13=a then ap=ap+1
2370    next
2380   }
2390    while bc=0
2400     symbol(176,48,"出したいカードを",1,1,1,15,0)
2410     symbol(208,84,"クリック",1,1,1,15,0)
2420     mouse(1)
2430     msarea(49,401,502,495):setmspos(64,432)
2440     repeat
2450      msstat(x,y,l,r)
2460     until l=-1 or r=-1
2470     mspos(x,y)
2480     mouse(0):er_upms()
2490   if m(3)>6 then is=(x-48)¥36
2500   if m(3)<7 then is=(x-48)¥58
2510   if is>m(3)-1 then {
2520     dame():wait(40):er_upms():continue}
2530      if b1>0 and ap>0 and (c(3,is)-1)¥13<>a then {
2540       dame():wait(40):er_upms():continue
2550     } else bacd(3,is):b(3)=p(3,is):sute(3)=b(3):bc=1
2560  endwhile
2570    if b1>0 and ap=0 and p(3,is)<100 then b(3)=0
2580     switch bb
2590     case 1: b1=b(3):break
2600     case 2: b2=b(3):break
2610     case 3: b3=b(3):break
2620     case 4: b4=b(3)
2630     endswitch
2640      p(3,is)=0:c(3,is)=0
2650      m(3)=m(3)-1
2660      for i=0 to 8:cc(i)=c(3,i):pp(i)=p(3,i):next
2670       cdleft(is)
2680       for i=0 to 8:c(3,i)=cc(i):p(3,i)=pp(i):next
2690        fill(2,385,391,509,0)
2700        plcd()
2710     } else b(3)=0
2720 endfunc
2730 /* 各トリック毎の判定
2740 func jd1()
2750   int i,j,a
2760   dim int ba(3)
2770   for i=0 to 3:ba(i)=b(i):next
2780   m_play(5)
2790   for i=0 to 2
2800     for j=i+1 to 3
2810      if b(i)<b(j) then a=b(i):b(i)=b(j):b(j)=a
2820     next
2830   next
2840    for j=0 to 3
2850     if b(0)=ba(j) then kachi(j):m_play(5):wait(60):break
2860    next
2870     jj=j:er_ms()
2880     m(jj)=m(jj)+1
2890  if ms>0 then {
2900    symbol(204,175,nam(jj)+"が1枚取ります",1,1,1,1,0)
2910    wait(20)
2920    c(jj,m(jj)-1)=stc(ms-1):p(jj,m(jj)-1)=stp(ms-1):ms=ms-1
2930      if jj<3 then pluscom(jj,m(jj)):jun(jj) else plcd()
2940      decst(ms)
2950  }
2960     fill(161,208,391,312,0):sutecd():wait(40)
2970 endfunc
2980 /* 各ラウンド毎の判定
2990 func jd2()
3000 int i
3010 m_play(4)
3020 symbol(182,224,nam(jj)+" "+str$(m(jj))+"枚残り",2,3,1,5,0)
3030   ten(rd,jj)=m(jj)*100
3040   apage(2)
3050   symbol(rd*30+338,jj*24+48,str$(ten(rd,jj)),1,2,0,15,0)
3060   for i=0 to 3
3070    if m(i)=0 then {
3080      ten(rd,i)=0
3090      symbol(rd*30+350,i*24+48,str$(ten(rd,i)),1,2,0,15,0)
3100    }
3110   next
3120   for i=0 to 3
3130     kei(i)=kei(i)+ten(rd,i)
3140     fill(481,i*24+49,509,i*24+71,10)
3150     if kei(i)<100 then {
3160       symbol(500,i*24+48,str$(kei(i)),1,2,0,15,0)
3170     } else  symbol(488,i*24+48,str$(kei(i)),1,2,0,15,0)
3180   next
3190   apage(1)
3200   rd=rd+1
3210   if rd<5 then {
3220    symbol(176,40,str$(rd)+"回目を始めます",1,1,1,15,0)
3230    s=sel(176,96,2,2)
3240    if s=2 then f=1:rd=5 else er_upms()
3250   } else rd=5
3260   apage(1):fill(0,0,511,511,0)
3270   apage(0):fill(0,144,511,511,0):apage(1)
3280 endfunc
3290 /* 最終判定
3300 func jd3()
3310   int i,j,a,b
3320   if f=0 then {          /*途中でやめた場合表示しない
3330    vpage(9)
3340    apage(0):fill(0,0,511,511,0)
3350    for i=0 to 5
3360      box(48+i*6,80+i*6,464-i*6,432-i*6,15)
3370    next
3380    fill(79,111,433,401,2)
3390    kei(3)=kei(3)+1
3400    for i=0 to 3:cc(i)=kei(i):next
3410     for i=0 to 2
3420       for j=i+1 to 3
3430        if cc(i)<=cc(j) then {
3440         a=cc(i):cc(i)=cc(j):cc(j)=a}
3450       next
3460    next
3470     for i=0 to 3
3480     if cc(0)=kei(i) then jj=i:break
3490     next
3500  symbol(97,218,nam(jj)+"の 勝ち!",2,2,2,5,0):m_play(7)
3510  symbol(352,440,"もう1度やりますか",1,1,1,15,0)
3520      s=sel(380,465,2,2)
3530       if s=1 then {
3540         rd=1:fill(0,0,511,511,0)
3550         apage(1):fill(0,0,511,511,0)
3560         apage(2):fill(0,0,511,511,0)
3570         vpage(15)
3580      } else f=1
3590   }
3600 endfunc
3610 /* 終了後の表示
3620 func owari()
3630    vpage(2):apage(1)
3640    fill(0,0,511,511,2)
3650    symbol(272,400,"お疲れ様でした",1,1,2,15,0)
3660    m_play(8)
3670 endfunc
3680 /*最弱の札を見付ける
3690 func weakest(jj)
3700   int i,j,a,is
3710   dim int p1(8)
3720   for i=0 to m(jj)-1
3730     if p(jj,i)<100 then p1(i)=p(jj,i)-1 mod 13 else {
3740      p1(i)=(p(jj,i)-101 mod 13)+100 }
3750   next
3760   for i=0 to m(jj)-2:for j=i+1 to m(jj)-1
3770     if p1(i)>p1(j) and p1(j)>0 then {
3780       a=p1(i):p1(i)=p1(j):p1(j)=a}
3790   next:next
3800 for i=0 to m(jj)-1
3810  if p1(0)<100 then {
3820  if p(jj,i)-1 mod 13=p1(0) then is=i:break
3830  } else if (p(jj,i)-101 mod 13)+100=p1(0) then is=i:break
3840 next
3850  return(is)
3860 endfunc
3870 /*comカードを強さでならべかえ
3880 func jun(k)
3890  int i,ii,a
3900  for i=0 to 7
3910   for ii=i+1 to 8
3920    if p(k,i)>p(k,ii) and p(k,ii)>0 then {
3930      a=p(k,i):p(k,i)=p(k,ii):p(k,ii)=a
3940      a=c(k,i):c(k,i)=c(k,ii):c(k,ii)=a
3950    }
3960   next
3970  next
3980 endfunc
3990 /*
4000 func scom2()
4010  int i,is,bc=0
4020    for i=0 to m(jj)-1
4030      if p(jj,i)>100 then is=i:bc=1:break
4040    next
4050    if bc=0 then is=weakest(jj)
4060  return(is)
4070 endfunc
4080 /*
4090 func scom3()
4100  int i,is,bc=0
4110    for i=0 to m(jj)-1
4120      if p(jj,i)>b2 then is=i:bc=1:break
4130    next
4140    if bc=0 then is=weakest(jj)
4150    return(is)
4160 endfunc
4170 /*
4180 func scom4()
4190  int i,is,bc=0
4200  for i=0 to m(jj)-1
4210   if p(jj,i)>b2 and p(jj,i)>b3 then is=i:bc=1:break
4220  next
4230  if bc=0 then is=weakest(jj)
4240  return(is)
4250 endfunc
4260 /*
4270 func sscom2()
4280 int i,is,a,bc=0
4290 if b1>100 then a=(b1-101)¥13 else if b1>0 then a=(b1-1)¥13
4300 for i=0 to m(jj)-1
4310  if (c(jj,i)-1)¥13=a and p(jj,i)>b1 then is=i:bc=1:break
4320 next
4330   if bc=0 then {
4340    for i=0 to m(jj)-1
4350     if (c(jj,i)-1)¥13=a then is=i:break
4360    next
4370   }
4380  return(is)
```

▶推薦で大学が決まりました。今年の正月から封じ込めていたX68000を解禁できます。いまいちばん欲しいゲームはやはり「スターウォーズ」ですね。　小林　順一(18)千葉県

```
4390 endfunc
4400 /*
4410 func sscom3()
4420 int i,is,a,bc=0
4430 if b1>100 then a=(b1-100)¥13 else if b1>0 then a=(b1-1)¥13
4440  for i=0 to m(jj)-1
4450   if (c(jj,i)-1)¥13=a and p(jj,i)>b1 and p(jj,i)>b2 then {
4460    is=i:bc=1:break}
4470  next
4480  if bc=0 then {
4490   for i=0 to m(jj)-1
4500    if (c(jj,i)-1)¥13=a then is=i:break
4510   next
4520  }
4530  return(is)
4540 endfunc
4550 /*
4560 func sscom4()
4570 int i,is,a,bc=0
4580 if b1>100 then a=(b1-100)¥13 else if b1>0 then a=(b1-1)¥13
4590  for i=0 to m(jj)-1
4600  if (c(jj,i)-1)¥13=a and p(jj,i)>b1 and p(jj,i)>b2 and p(j
j,i)>b3 then {
4610  is=i:bc=1:break}
4620  next
4630  if bc=0 then {
4640   for i=0 to m(jj)-1
4650    if (c(jj,i)-1)¥13=a then is=i:break
4660   next
4670  }
4680  return(is)
4690 endfunc
4700 /*
4710 func cdleft(k)
4720  int i
4730  for i=0 to 8-k:cc(k+i)=cc(k+i+1):pp(k+i)=pp(k+i+1):next
4740 endfunc
4750 /*
4760 func sel(x,y,m,n)
4770  int i,j,a,b
4780  str mm,nn
4790  switch m
4800   case 1:mm="必　要":break
4810   case 2:mm="O　K"
4820  endswitch
4830  switch n
4840   case 1:nn="不　要":break
4850   case 2:nn="やめる"
4860  endswitch
4870  fill(x,y,x+56,y+24,15):fill(x+72,y,x+128,y+24,15)
4880 symbol(x+4,y+4,mm,1,1,1,1,0):symbol(x+76,y+4,nn,1,1,1,1,0)
4890  mouse(1)
4900  msarea(x+1,y+1,x+127,y+23)
4910  setmspos(x+28,y+8)
4920  repeat
4930   msstat(i,j,a,b)
4940  until a<>0 or b<>0
4950   mspos(i,j)
4960   mouse(0)
4970   if i<x+64 then {
4980   fill(x,y,x+56,y+24,1):symbol(x+4,y+4,mm,1,1,1,15,0):s=1
4990   } else {
5000  fill(x+72,y,x+128,y+24,1):symbol(x+76,y+4,nn,1,1,1,15,0)
5010   s=2}
5020   return(s):wait(40)
5030 endfunc
5040 /*
5050 func click()
5060  int i,j,a,b
5070  fill(168,40,312,108,4)
5080  symbol(176,48,"よければマウスを",1,1,1,15,0)
5090  symbol(208,84,"クリック",1,1,1,15,0)
5100  mouse(1)
5110  msarea(176,48,288,96)
5120  setmspos(232,70)
5130  repeat
5140    msstat(i,j,a,b)
5150  until a<>0 or b<>0
5160  mouse(0)
5170  er_upms()
5180 endfunc
5190 /*
5200 func mkba()
5210  int i
5220  apage(2)
5230  fill(161,145,391,351,8):fill(184,164,368,200,15)
5240  fill(393,145,509,319,8):fill(393,321,509,509,8)
5250  fill(161,353,391,383,6)
5260  symbol(396,164,"ストックカード",1,1,1,15,0)
5270  symbol(412,336,"捨　て　札",1,1,1,15,0)
5280  for i=0 to 2:symbol(8,i*100+112,nam(i),1,1,1,15,0):next
5290  symbol(8,428,nam(3),1,1,1,15,0)
5300  for i=0 to 3:symbol(i*55+177,316,nam(i),1,1,1,15,0):next
5310  apage(1)
5320 endfunc
5330 /*
5340 func dsban(j)
5350   er_ms():symbol(212,175,nam(j)+"の番",2,1,1,1,0)
5360 endfunc
5370 /*
5380 func kachi(jj)
5390   er_ms():symbol(200,175,nam(jj)+"の勝ち",2,1,1,5,0)
5400 endfunc
5410 /*
5420 func dame()
5430 er_upms():symbol(180,60,"出せません",1,1,2,5,0):m_play(3)
5440 endfunc
5450 /*
5460 func bacd(j,i)
5470   c_put(j*55+169,216,c(j,i)):m_play(1,2)
```

```
5480 endfunc
5490 /*プレイヤ・カードの表示
5500 func plcd()
5510  int i
5520  fill(47,400,386,496,0)
5530  if m(3)>6 then {
5540   for i=0 to m(3)-1
5550    c_put(i*36+48,400,c(3,i))
5560    line(i*36+47,400,i*36+47,496,10):m_play(1,2)
5570   next
5580   } else {
5590    for i=0 to m(3)-1
5600    c_put(i*58+48,400,c(3,i))
5610    next
5620   }
5630   fill(16,453,24,469,0)
5640   symbol(16,453,str$(m(3)),1,1,1,15,0)
5650 endfunc
5660 /*comカードの表示
5670 func comcd(j)
5680  int i
5690  fill(47,j*100+84,153,j*100+180,0)
5700  for i=0 to m(j)-1
5710   c_put(i*7+48,j*100+84,0)
5720   line(i*7+47,j*100+84,i*7+47,j*100+180,10):m_play(1,2)
5730  next
5740  fill(16,j*100+136,24,j*100+152,0)
5750  symbol(16,j*100+136,str$(m(j)),1,1,1,15,0)
5760 endfunc
5770 /*comカード1枚追加の表示
5780 func pluscom(j,m)
5790  c_put((m-1)*7+48,j*100+84,0)
5800  line((m-1)*7+47,j*100+84,(m-1)*7+47,j*100+180,10):m_play(
1,2)
5810  fill(16,j*100+136,24,j*100+152,0)
5820  symbol(16,j*100+136,str$(m),1,1,1,15,0)
5830 endfunc
5840 /*ストックカードの表示
5850 func stcd()
5860  int i
5870  for i=0 to ms-1
5880   c_put(i*4+399,192,0)
5890   line(i*4+398,192,i*4+398,288,8):m_play(1,2)
5900  next
5910   fill(444,292,460,308,0)
5920   symbol(444,292,str$(ms),1,1,1,15,0)
5930 endfunc
5940 /*ストックカード1枚減らす表示
5950 func decst(i)
5960  fill(i*4+398,192,i*4+447,288,0)
5970   if i>0 then c_put(i*4+395,192,0):line(i*4+394,192,i*4+39
4,288,8)
5980     m_play(1,2)
5990     fill(444,292,460,308,0)
6000  symbol(444,292,str$(i),1,1,1,15,0)
6010 endfunc
6020 /*捨て札の表示
6030 func sutecd()
6040  int i,x,y
6050  for i=0 to 3
6060   if sute(i)>100 then sute(i)=sute(i)-100
6070   if sute(i)=13 or sute(i)=26 or sute(i)=39 or sute(i)=52
then {
6080    sute(i)=sute(i)-12
6090   } else  sute(i)=sute(i)+1
6100   x=rnd()*56+400:y=rnd()*44+364
6110   c_put(x,y,sute(i)):box(x-1,y-1,x+47,y+95,8):m_play(1,2)
6120  next
6130 endfunc
6140 /*
6150 func wait(t)
6160  int i
6170  for i=0 to t*100:next
6180 endfunc
6190 /*
6200 func er_upms()
6210  fill(161,3,319,143,0)
6220 endfunc
6230 /*
6240 func er_ms()
6250  fill(184,164,368,200,15)
6260 endfunc
6270 /*
6280 func rule()
6290 apage(0)
6300 fill(2,145,509,383,12)
6310 line(2,144,159,144,15):line(391,384,509,384,15)
6320 symbol(196,160,"ル　ー　ル",1,1,1,15,0)
6330 symbol(60,182,"1 :　カードの強さは A,K,Q,J...4,3,2 の順",1,1,1,15
,0)
6340 symbol(60,200,"2 :　貴方に最後に配られたカードのスーツが切り札",1,1,1,15,
0)
6350 symbol(60,218,"3 :　各自1枚づつ最初と同じスーツを出さねばなりません",1,1,1
,15,0)
6360 symbol(60,236,"4 :　持ってなければ切り札又は他のカードを出します",1,1,1,1
5,0)
6370 symbol(60,254,"5 :　最強の切札又は台札と同じスーツの最強のカード",1,1,1,1
5,0)
6380 symbol(92,271,"を出した人が勝ち",1,1,1,15,0)
6390 symbol(60,289,"6 :　勝った人はストックから1枚取り手札に加えます、また",1,1
,1,15,0)
6400 symbol(92,306,"次は最初に台札を出します",1,1,1,15,0)
6410 symbol(60,324,"7 :　手札が無くなった人は負け、もうプレー出来ません",1,1,1,
15,0)
6420 symbol(60,342,"8 :　全員が脱落した後まだカードを持っている人が勝ち",1,1,1,
15,0)
6430 symbol(60,360,"9 :　最後に持っているカード1枚に付き100点、他は0点",1,1
,1,15,0)
6440 apage(1)
6450 endfunc
```

# 愛読者プレゼント

PRESENT

## 1

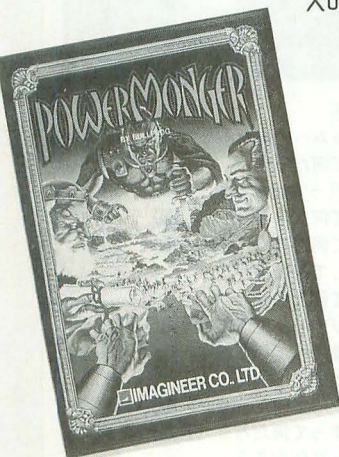
イマジニア ☎03(3343)8911

### パワーモンガー

X68000用 5"2HD版

12,800円(税別)

2名

あのピーター・モリニュー氏が放つ3Dリアルタイムシミュレーションゲーム。AMIGA版からの移植だ。

## 2

ファミリーソフト ☎03(3924)5727

### 機動戦士ガンダム クラシックオペレーション

X68000用 5"2HD版3枚組

9,800円(税別)

3名

ひとつのシナリオが1〜2時間で終えられるお手軽シミュレーション。ガンダムファンお待ちかね(?)の1本だ。

## 3

シャープ ☎03(3260)1161

### ダウンタウン熱血物語

X68000用 5"2HD版2枚組

8,800円(税別)

3名

くにおくんシリーズ第3弾。不良対不良のケンカアクションゲームだ。道端に転がっている鉄パイプやポリバケツまで武器になるのが楽しい。

## 4

アルシスソフトウェア ☎0956(22)3881

### スピンディジーII ボールペン

5名

アルシスひさびさの新作スピンディジーII。その発売を記念して作られた3色ボールペンを，ブラック，グレーの2本セットで。

## 5

キングレコード ☎03(3945)2122

### 出たな!! ツインビーCD

2,800円(税込)

3名

もう移植ゲームのほうも発売されたけど，こちらのCDのほうもなかなかだ。アレンジバージョン，サウンドエフェクトを含む全22曲を収録。

### プレゼントの応募方法

とじ込みのアンケートはがきの該当項目をすべてご記入のうえ，希望するプレゼント番号をはがき右下のスペースにひとつ記入してお申し込みください。締め切りは1992年1月18日の到着分までとします。当選者の発表は1992年3月号で行います。

### 11月号プレゼント当選者

1ロードス島戦記（群馬県）澤英敏（大阪府）野村友彦（京都府）矢野裕明 2ボナンザブラザーズ（北海道）横溝貴志（神奈川県）田辺和也（兵庫県）中垣敦 3エイトレイクスゴルフクラブ（東京都）萩原今朝巳（神奈川県）森田真史（京都府）松永正弘 4ニンジャウォーリアーズCD（三重県）松居昭宏（奈良県）西岡山誠人 5上昇気流Vol.2（茨城県）広瀬良一（千葉県）江ヶ崎貞行（神奈川県）飯島徹（広島県）的場全（福岡県）竹岡均ほか5名 （敬称略）

以上の方々が当選しました。おめでとうございます。商品は順次発送いたしますが，入荷状況などにより遅れる場合もあります。また，雑誌公正競争規約の定めにより，このプレゼントに当選された方は，この号の他の懸賞には当選できない場合がありますのでご了承ください。

# X68000 C プログラミング

●中森 章

●B5変形判・340ページ●定価2600円（税込）

■本書の内容

1 プログラムって何だろう
2 変数って何だろう
3 制御構造って何だろう
4 配列って何だろう（一次元編）
5 配列って何だろう（多次元編）
6 文字列って何だろう
7 関数って何だろう
8 再帰呼び出しって何だろう
9 式と演算子って何だろう
10 標準入出力って何だろう
11 ポインタって何だろう（前編）
12 ポインタって何だろう（後編）
13 構造体って何だろう
14 ファイル入出力って何だろう
15 プリプロセッサって何だろう

**X68000上でのCによるプログラミングをXCの利用を中心に初歩からわかりやすく解説。プログラムを書く際どのような点に注意すべきか，C言語に用意されている機能はどう活用したらよいのかを豊富な実例，設問を交え紹介した。軽妙な語り口で好評を得た『Oh!X』誌連載「ようこそここへC言語」の書籍化。**

## 好評既刊

### X68000マシン語プログラミング入門編

村田敏幸著

●B5変形判●定価2800円

『Oh!X』の好評連載をまとめた単行本。プログラミングの力は実際にプログラミングするなかから培われるという視点で，豊富な実例を示しながら，マシン語プログラミングのおもしろさを解説。

### SX-WINDOW プログラミング

吉沢正敏著

●B5変形判●定価2800円

X68000にマルチタスク，マルチウィンドウ環境をもたらしたSX-WINDOW上でプログラミングするにはどうすればいいか。著者独自の内部解析にもとづいたプログラミングの実例を示す。

## 近刊予定

### 12月 追補版SX-WINDOWプログラミング

吉沢正敏著

●B5変形判●予価3800円（5インチFD付き）

SX-WINDOW ver.1.10で新たに加わったマネージャ，SXコールなどを解説。付録ディスクには，本書で取り上げたサンプルプログラム以外に，ver.1.10対応のCのライブラリ（サンプル版）と，PDSとして公開されているSX-WINDOW用のプログラムを収録。

### 1月 X68000マシン語プログラミング グラフィックス編

村田敏幸著

●B5変形判●予価3500円（5インチFD付き）

入門編に引き続き，『Oh!X』誌に連載されたもののうち，グラフィック関連の連載をまとめたもの。付録ディスクには，本文中で取り上げたプログラムのほかに，著者が新たに作成したプログラムも収録。

### 2月 インサイドX68000

桒野雅彦著

●B5変形判●予価3500円

使用者の立場に立った，使いやすいX68000のハードウェア解説書の決定版。ハードディスクインタフェイスや，AD PCMなど，これまでの解説書では取り上げられなかった部分についても詳説。

### 2月 GNUツールボックス

吉野智興著

●B5変形判●予価3600円

GNUツールはX68000にいかに移植されたか？ GNUツール（GCC, G++, Nemacs）をX68000に移植する際の方法とノウハウを，実際の移植者が豊富な実例を挙げて明快に解説。<G++を収録した5″2HD FD付き>

●問い合わせ ソフトバンク㈱出版事業部 ☎03-5488-1360

# ノイマンはなぜノイマンマシンを作ったのか？

## フォン・ノイマン型計算機

フォン・ノイマン型計算機ということばがあります。このことばは実はそれほどしっかりと定義された用語とはいえないのですが，それでも世の中に出回っている計算機のほとんどはこの種類に属するといっても間違いとはいえないでしょう。

フォン・ノイマン型計算機の特徴を簡単にまとめてみると次のようになります。

1) プログラムはデータと同様にメモリに入れる（メモリ格納型プログラム）
2) メモリには，0番地，1番地というように1次元的にアドレスが付けられている(線形アドレス)
3) メモリから順番に命令をひとつずつ取ってきて実行する（逐次実行）

「なんだ，当たり前じゃないか」と思う人も多いかもしれません。確かにいまでは当たり前となっていることです。しかし，フォン・ノイマンがこれを提案する以前にはソフトウェアという概念はなかったのだ，という事実を挙げるだけでも，フォン・ノイマンの業績をわかってもらえるようになるかもしれません。それまでは，プログラムとはスイッチの設定でただ計算方式を指定することにすぎなかったのですから。

フォン・ノイマン型計算機のごく基本的な構造を図1に示します。計算機はCPUとメモリから構成されています。CPU内のプログラムカウンタ(PC)の値は，次に実行すべき命令の入っているメモリアドレスを表しています。この値をメモリに送りつけると，メモリはそのアドレスの中に入っている命令を送り返してきます。そしてCPUはその命令に応じた処理（レジスタの中身のコピー，加算，減算，あるいはレジスタ＝メモリ間でのデータ転送など）を行います。

通常，命令は実行順に格納されているので，プログラムカウンタの値を順に増やしていくだけで(分岐命令は別)，あとは同じようにして，逐次的に処理が進んでいきます。このように実に簡単な機構でフォン・ノイマン型計算機は構成されています。現在の圧倒的な繁栄もこのようなシンプルさゆえのものでしょう。

## フォン・ノイマン型計算機の限界

フォン・ノイマン型計算機に対する問題点はいろいろと指摘されています。中心的なのは，やはりなんといっても性能の限界が見えているということでしょう。その具体的な問題のひとつに，「フォン・ノイマン・ボトルネック」と呼ばれているものがあります。

図1を見ても，CPUとメモリをつなぐ経路が1個命令を実行するたびに頻繁に使われるということがおわかりになるでしょう。1命令を実行するたびに，命令，アドレス，データなどを通さなければならないのです。そして，この経路が混雑しすぎてしまうと，いくらCPUの速さを上げても待たされてしまい，CPU改良の意味がないという事態が発生してしまうのです。

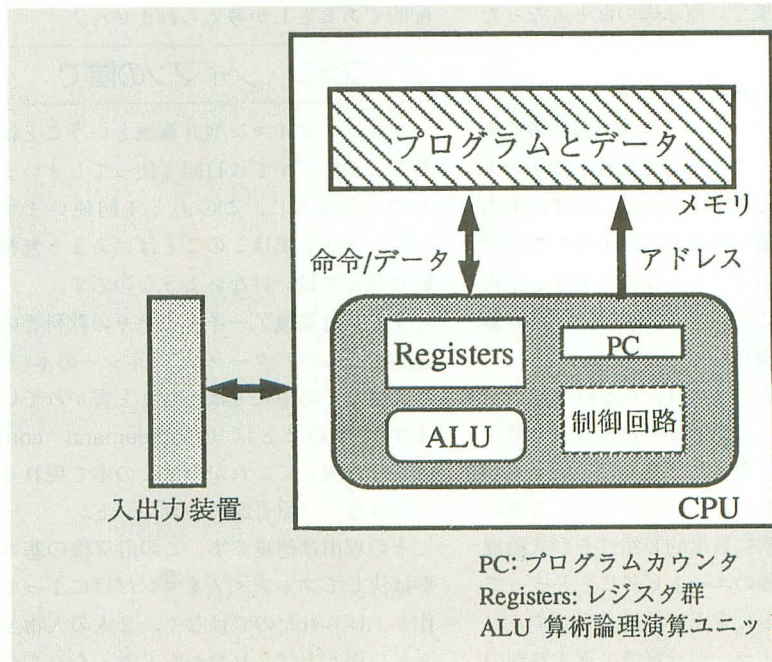
このボトルネックを解決するには，この経路の幅や速さ（バンド幅）を広げるか，この経路を使う量を減らすように，根本的な部分を変更するしか手はありません。しかし，前者には（これからますます重要となる）CPUの1チップ化において，ピン数などの物理的制限が目の前に立ちはだかりますし，後者もレジスタ数を増やしたり別の高速メモリを導入したりしていますが，根本的な解決には至ってはいないようです。

確かに，このようなフォン・ノイマン・ボトルネックは高速化する際の大きな問題のひとつではあります。しかし，根本にあるのは別の問題であると考えることができます。それは，前述した特徴の3番目にある逐次処理ということです。

いくらひとつのプロセッサががんばっても限界があります。大勢が力を合わせる並列処理にかなうわけはないのです（と思って研究をしている）。

## 天才フォン・ノイマン

逐次処理あっての並列処理ですから，こんな質問はナンセンスともいえましょうが，

図1　フォン・ノイマン型計算機の基本構成

あえて次の疑問を考えてみることにしましょう。「フォン・ノイマンはなぜ並列処理ではなく逐次処理を行う計算機を提案したのでしょうか？」

とりあえずの答えとしては，フォン・ノイマンは，(今日すべての計算機研究者が思っているように）並列処理型計算機は難しいこと，あるいは，逐次型計算機の改良(命令パイプラインとかRISCなど)により十分性能が出るということをすでに直観的に理解していて，並列処理をあきらめてきわめてシンプルな逐次処理にしたとこじつけることも可能といえば可能でしょう。

しかし，ここで思考停止しないで，フォン・ノイマンという人物，あるいは彼の生きた時代に即して，もう少し考えてみることにしましょう。彼について触れられている興味深い文化論（参考文献１）もあることですし。

ジョン・フォン・ノイマン（1903-1957）は，単にフォン・ノイマン型計算機の原形を示しただけ（などといったら天罰が下りそう）でなく，オートマトンやゲーム理論の基盤を作り，量子力学を数学的に体系化し，挙げ句の果てに原水爆の父ともなった人です。

西垣氏のこの著書の中では，フォン・ノイマンを，サイバネティクスの創始者であるノーバート・ウィーナと比較してその人物像を浮き立たせようとしています。もちろん，ともに情報科学の基礎を築いた天才的な研究者であることには違いありませんが，大それたことにフォン・ノイマンの「器の小ささ」を指摘しています。

器の小ささというのは，彼が時と場所を心得て，愛想よくふるまったり威張り散らしたりするような人物であったという意味でもあるのですが，それよりはむしろ彼の研究や関心が常に数学的必然性や形式論理に囚われた一種のユートピアにとどまっていたということに重点があるようです。

フォン・ノイマン型計算機の逐次処理の数学的な背景とはなんなのでしょうか？それが万能チューリングマシンです。アラン・チューリング(1912-1954)が考え出した，０か１のどちらかが書かれたテープとそれを読むことのできるヘッドを持った順序機械からなるこの簡単なモデルは，すべての記号操作，論理操作が可能であると広く知られており，もちろんそのことは数学的に示されています。

もう，ほとんど書いてしまったも同然ですね。フォン・ノイマン型計算機は逐次処理で（彼にとっては）十分だったというわけです。彼の数学的宇宙において，すでにそれは万能であったのですから。

誤解を避けるためにひと言付け加えるのならば，数学的宇宙の外においても，逐次処理に本質的な部分があるはずだということの状況証拠はあります。そのひとつとして，人工知能が目指している人間の思考そのものが挙げられます。人間の思考においては，逐次処理はかなりのウエイトを占めていると考えられます。「おなかがすいたから食事をして，それから……」などと，少なくとも意識の表層では逐次的な処理が支配的であるとしか考えられません。

## フォン・ノイマンの陰で

フォン・ノイマン型計算機ということばをここまでですでに11回も使ってしまいました（ちなみに，このあと４回使いますが)。しかし，実はこのことばはあまり無神経に使ってはいけないようなのです。

すでに計算機アーキテクチャの教科書の定番となったパターソン，ヘネシーの本(参考文献２）の中にもはっきりと書かれています。「このことば（vonNeumann computer)は決してこれからはこの本で現れることはない（原書24ページ)」，と。

その理由は簡単です。この計算機の基本形は決してフォン・ノイマンだけによって作り上げられたのではなく，２人の人物とともに作り上げられたからと考えられるからです。その２人とは，J.P.EckertとJ.Mauchlyです。彼らは汎用の(スイッチによってプログラムする)最初の計算機ENIACを作り上げた人たちです。

フォン・ノイマンは彼らのグループに興味を持ち，プログラムをメモリに置くタイプの計算機のアイデアをメモ書きしました。しかし，このアイデアの原形はすでにこのグループで議論されていたのです。それを別の人がフォン・ノイマンの名前を書き添えて配った，というのがこの名前のそもそもの由来なのです。

したがって，エッカート・モーリ・ノイマン型計算機と称するならば許されるのでしょうが，フォン・ノイマン型計算機というのでは，あとの２人に対して公平でないという主張があるのです。確かにもっともといえばもっともな話ですね。

## パラレル・ノイマン型計算機

フォン・ノイマン型計算機をなるべく速く動かすために，並列実行化したいという要求は大きいものがあります。それに応えようとするアプローチにもいろいろありますが，プロセッサレベルの並列化でもっとも成功しているのが命令パイプライン化といえるでしょう。要するにひとつの命令をプログラムからとってきて解読し，実行するという一連の処理を流れ作業的に行うことにより，高速化しようというものです。

そのようなアプローチとは一線を画し，もっと基本的なところからの並列化を目指しているのが，我々が提案している並列実行モデル（参考文献３，４）です。パラレル・ノイマン型計算機とも呼んでいますが，これについて少しだけ紹介することにしましょう。

その基本構成を図２に示します。実にシンプルなものです。図１では実行ユニットがCPU１個でしたが，こちらではそれを複数にすることができます（３つに増やした場合を例にしています)。最大５倍の高速化

# ノイマンはなぜノイマンマシンを作ったのか?

をしたいのならばPUを5個にします。なお，この図ではPU内にはPCだけ書いており，ALUやレジスタなどは省略しています。

重要なのは，各PU間にトークンカウンタ(TC)と呼んでいる通常のカウンタを2個ずつ接続しているところです。これによって，並列実行がうまくいくようにしているのです。

PU1，PU2，PU3はそれぞれプログラムカウンタを持っており，フォン・ノイマン型計算機と同じように命令を実行します。ただし，別のPU間で実行される命令間にときどき，実行の順番に関する制約(先行制約と呼ぶ)が課せられることになります(その必要がないのならば，図1を3つ寄せ集めたのと変わりがなくなってしまいます)。たとえば，PU1で演算結果をメモリに格納し，PU2がその結果を読み込むなどなど。

このように特定の2命令間での先行制約が定められたとおりに守られるようにするための機構がトークンカウンタなのです。プログラムカウンタがそれぞれのPUの命令実行の進み具合を命令の入っているメモリアドレスで表すのに対し，トークンカウンタはそれぞれの命令の実行の進み具合の差をPU間にまたがる先行制約によって表しているのです。

メカニズム的にいうならば，ほかのPUの命令と歩調を合わせなければならない命令にはあらかじめ印を付けておき，その命令を実行する直前や直後にトークンカウンタを操作(1増やしたり，1以上になるまで待つ)するようにすればよいのです。

## 数学と一線を画すべきか

今回の話をなぞると，「フォン・ノイマンは数学世界にしか興味がなかった→数学的に必要十分なノイマン・マシンを作った→逐次型なので性能に不十分となり並列化の研究をして(苦しんで)いる」となるでしょうか。

この筋道は素直に受け取ると，数学世界の中で閉じていることはあまりよくないことであると主張しているようにも思われますし，実際，参考文献1の著者は間接的にそのように書いています。

しかし，僕自身はそのようにはまったく思っていません。チューリング・マシンではあらゆる論理的計算が計算可能であるということが数学的に示されているのであり，逆にいえば実行ユニットを複数持たせるとか，性能がどうだとかは考慮に入れていないのです。

ところが，そのようなことを考慮に入れて，しかもきちんと科学的に話を進めることができるのもまた数学であると僕は思っているのです。

あいかわらず，計算機を研究している人の中には，数学は数学の好きな人にまかせて，という風潮が残念ながら強いようです。しかし，実はこれがフォン・ノイマンの呪縛から抜け出せない大きな理由となっていると思われます。

いまこそ，あまりに単純すぎるが十分に強力であるフォン・ノイマン型計算機の外面ではなく，なにがこれを作り上げたのかという基盤となっている強固な背景に目を向ける必要があるのではないかと思っています。

この世界には成仏できなかったペーパーマシン(紙の上の提案や設計に留まった計算機)の亡霊たちが無数にさまよっているのですから。


参考文献

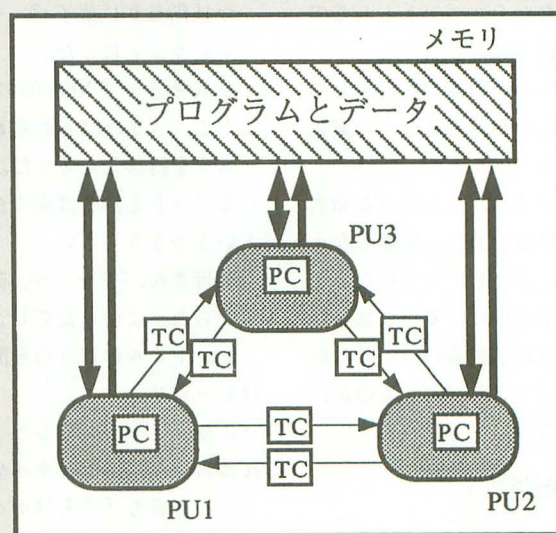
1) 西垣通，"デジタル・ナルシス"，岩波書店，1991.

2) D.A.Patterson and J.L.Hennessy, "Computer Architecture―A Quantitative Approach", Morgan Kaufmann Publishers, 1990.

3) T.Arita et.al., "High Speed Synchronization for a Statically-scheduled Superscalar Processor", International Journal of High Speed Computing, Vol.3, No.1, pp.77-87, 1991.

4) 高木 浩光 他，"問題が持つ先行関係のみを保証する高速な静的実行順序制御機構"，情報処理学会論文誌(1991-12).


図2 パラレル・ノイマン型計算機の基本構成

第66回

# 猫とコンピュータ

# ギャラガ for Xmas

Takazawa Kyoko
高沢 恭子

しばらく「Columns」にご執心のキョウコさんでしたが，いまのお気に入りは「ギャラガ'88」。1年以上前のゲームですが，キョウコさんはX68000版は初めてとあって，楽しくプレイしているようですが……。

お客さまの気配がすると，外に逃げるか押入れにかくれるのは，もちろんホンニャアだ。ときには，例の天才的な予知能力で事前にそれをやる。

おだやかにカラリと晴れた日曜日の真昼だというのに，ホンニャアはすでに押入れにしのびこんで丸くなっている。カンが良いのか眠いだけなのか。

押入れに長いこと寝込まれると，ふとんがなま暖かくへこんだあげく，遺留品の抜け毛がどっさりへばりつく。

こういうことは新宿の猫たちにはけっしてゆるされなかった。父母の家には，1年のうち半分くらいは，アンチ猫派の祖母がいたので，飼い猫は行動をすべてチェックされ，不衛生なことや汚れの原因になることは禁止されたものだ。

祖母がいないときでも，両親が勤めていたために，家にはいつも家事を手伝ってくれる若い女性がいたから，ペットとしての動物はきびしく監視されていた。

そのおかげか新宿の家の猫たちは，しつけらしいことも覚えたし，それなりに緊張感のある良い顔をしていた。

ホンニャアについては，フマジメな命名からして根本的な反省がある。風貌もなんとなくおめでたい。きびしいしつけなんて誰もしなかった。家の中にいるときはいいとしても，こんな彼が猫組織の中でどれほどの威厳と底力を示すことができるのか，あまり期待はできない。

## 昼メシ前の出会い

正午前，ホンニャアの予知した来客がおとずれた。トオルのクラスメート，フクシマ君とシバタ君だった。

高校生活で，いまいちばん楽しいと感じるのは，いろいろな友人と語り合うこと，そうしながらたくさんのことを考えることだと，トオルは言う。時間を忘れて友人と話をしているときが，最高の満足なのだそうだ。

実力テストが済んでちょっとひと息という，11月上旬の日曜日。おとずれた人たちにも，迎えるトオルにも喜色満面のかがやきがある。

1時間ほどトオルの部屋ですごしたあと，3人そろってマシンルームにやってきた。

「X68000，使っていい？」

「もちろん，ご自由にどうぞ」

イスを3つとお茶をサービスしながらよく聞くと，どうもフクシマ君もシバタ君も昼食をほとんどとらずに家を出てきたらしい。やっぱり！　高校生の日曜日にしては出足が早いと思った。でも，ゴハンなんて1，2回ぬいても，友人とすごすほうが楽しいという気持ちは，とてもよくわかる。

お昼代わりに菓子パンやクッキーを食べてもらうことにした。

2人そろってX68000のユーザーということで，きょうはそれぞれ所有のゲームディスクを持参したそうだ。トオルの持っているソフトと，しばらく交換して楽しもうというつもりらしい。

「お母さん，『ギャラガ』のX68000版を借りるからね。よかったでしょ！」

と，トオルの言うのを聞いてフクシマ君はビックリ。

「エッ？　ゲームやるの？　うちのお母さんは古いんだ，ゼッタイやらないよ」

シバタ君も，「うちはボクが何時間やってても，ひと言も言わないんだ」。

パソコンにあまりなじみのない人にとっては，パソコンゲームは純粋なアソビであって，けっして推奨できるようなものではないだろう。批判的に感じたり，静観の態度で接するのは，親の立場になればよくわかることだ。

逆に，パソコンを動かす習慣が少しでもある人なら，ゲームにもパソコンを使うのはごく自然なことだ。パソコンの特性や操作を好む人にとって，ゲームもパソコンライフの一環といえる。

ボタン操作の痛快さ，いながらの躍動感を満喫しつつも，ゲーム作者グループの手腕を賞味している気分もあるだろう。

第一線のプログラマのコマキ氏も，ゲームが大好きだそうで，20時間くらい眠らずにゲームに没頭しているのを見たと，やはり同業のナカニシ氏から聞いた。プロゴルファーが，息抜きにゴルフをやるかどうかは知らないけれど，まったくちがった状況下で自分の専門分野にふれるのも，思わぬインスピレーションを得るチャンスになるかもしれないと思う。

ところで，「Oh!X」の読者ではないフクシマ君とシバタ君だから「猫とコンピュータ」なんて夢にも知らない。X68000のゲームと聞いてうれしそうに寄ってくる友人のお母さんは，さぞめずらしい存在だろう。

## パーティグッズに

この日のX68000は，つぎからつぎへとゲームをロードされては，ためし撃ちをあびて，ドドーン，バキューンと大活躍。私が音もなく使うマシンとは別ものみたいにはなやかだった。

トオルは2人のお友だちから，5つのゲームを借りた。「ファンタジーゾーン」，「殺人倶楽部」，「パックマニア」，「琥珀色の遺言」，そして「ギャラガ '88」。

こちらからも，「ザ・リターン・オブ・イシター」，「グラディウス」，「パワーリーグ」，「サンダーフォースII」，「源平討魔伝」

が出陣していった。

ちかごろ「Columns」のほかは，あまりゲームのしごとをしていないX68000だが，この日をきっかけに元気なゲームマシンの顔を取り戻した。

「ギャラガ」のオリジナル版は1981年にナムコが制作したもので，はじめて出会ったのはつい３年くらい前だった。そのころちょっと話題になっていた，ゲームセンターなどの中古基盤のリフォームに，夫がトライしてみたのだが，その年の「ホビーショウ」にも陳列して，来場者に楽しんでもらった。

それ以後，この「再生ギャラガ」は，わが家のマシンルームで専用のモニタテレビと指定席をあてがわれ，常駐のゲーム機となってサービスにつとめていた。

その間，1987年に本家の「ギャラガ」はバージョンアップ，「ギャラガ '88」となったらしい。それを1990年に電波新聞社がX68000用に移植したものが，今回フクシマ君から借りた「ギャラガ '88」というしだいだ。

空中に飛び交う敵を攻撃機のミサイルでつぎつぎ撃ち落とし，各面をクリアしていくという，シンプルなシューティングゲームだった元祖「ギャラガ」は，10年で大変貌していた。

ゲームの構成が何層にもなり，ステージとなる宇宙空間も，ぐんと大きく広がって迫力がまるっきりちがう。音響効果がそれをさらに深めて，立体感も倍増。それというのも，X68000の高画質とFM音源ゆえだろうが，この中で飛び回る敵の軍団もすっかり豪華になり，数の豊富なこと，生気にあふれハツラツとしていることなど，びっくりするばかりだ。

蛾と思われる虫のさまざまなバリエーション。花，星，隕石らしいもの。アヒルや，タコやトンボにしか見えないもの。奇抜できらびやかで，カラフルなキャラクターが，意表をついた変身をつぎつぎに見せながら，ミサイルの攻撃をあびると，花火のような痛快な音をあげてくだけ散る。

各ステージを，順にクリアして進んでいくのは変わらないが，次元(DIMENSION)をワープすることもでき，得点は急増するけれど，敵の強さが増して難易度も上昇する。

うれしいのは，継続(CONTINUE)の設定をしておくと，攻撃機が全滅してもその状態からゲームを続行できることで，むずかしい次元や面もかならずクリアして，新しいステージで愉快な敵とつぎつぎ交戦できる。ワザに自信がなくても，スリル満点の銀河の旅を，不死身で思いっきり楽しめるという，遊園地のようなゲームだ。

そうだ，見晴らしのよい銀河と，奇怪で愛きょういっぱいの敵軍団のオールスターズ，連発する花火の音は，クリスマスの夜にふさわしい。ボーナスステージではワルツやタンゴのサービスもある。ことしのイブの出し物の中に「ギャラガ '88」を加えよう。

## 虫のタタリじゃ！

眠っていたX68000用のゲームソフトたちは，おたがいの家に派遣されて，目をさました。「パックマニア」，「ファンタジーゾーン」のキャラクターにも，X68000の画面で再会できるとは思わなかった。「源平討魔伝」を持ち帰ったシバタ君も，同じ思いかもしれない。

「琥珀色の遺言」は殺人推理劇のAVGで，シリアスなグラフィックとサウンドに念を入れてあった。

しまわれていたままの衣類に，風や日光を通すのは「虫干し」だが，蛾の大群が舞いおどる「ギャラガ '88」は，まさに秋の虫干しになっただろう。エンディングは難易度別に４つくらいあるらしい。いちばん深い５次元では，体長５センチくらいの蛾のボスが登場して，クリアするとかわいい少女があらわれ，栄誉をたたえてくれる。勝利のアカシに，ボロボロになったボスの姿もさらされる。

貯蔵が長くなり，活用の機会に期待のもてないものは，死蔵品と呼ばれるようになるけれど，パソコンの世界は，いつもソフトとハードの死蔵品の山頂を歩いているようにも見える。死蔵品とはちがうけれど，貴重なデータやプログラムは，意外なところにびっしりと貯蔵されているものだ。

先日，NIFTY-Serve(パソコン通信ネット）の階層構造の一部である，いくつかのボードや会議室の，登録内容のタイトル総覧を見ておどろいた。

オンラインソフトウェア（PDSといわれていたもので，おおやけに提供されたプログラム，有料と無料がある）のボードに，X68000に関するプログラムだけでも，あらゆる種類のものが，それこそ銀河のように並んでいた。

illustration Kyoko Takazawa

こんなにたくさんのプログラムの山は，項目の列を見ただけでエネルギーの凝縮を感じて熱くなる。プログラムだから，いくら蓄積しても公害にはならないけれど，ふと，日本じゅうの，世界じゅうのデータベースの海を思い浮かべて，目まいがしそうになった。

マシンの機種もどんどん更新されていくことだし，プログラムはあれよあれよという間に改良されるだろう。利用されるより生産されるほうが多ければ，蓄積過剰になるのはあたりまえ。プログラムばかりでなく，あらゆるジャンルの情報だって同じようなものだ。

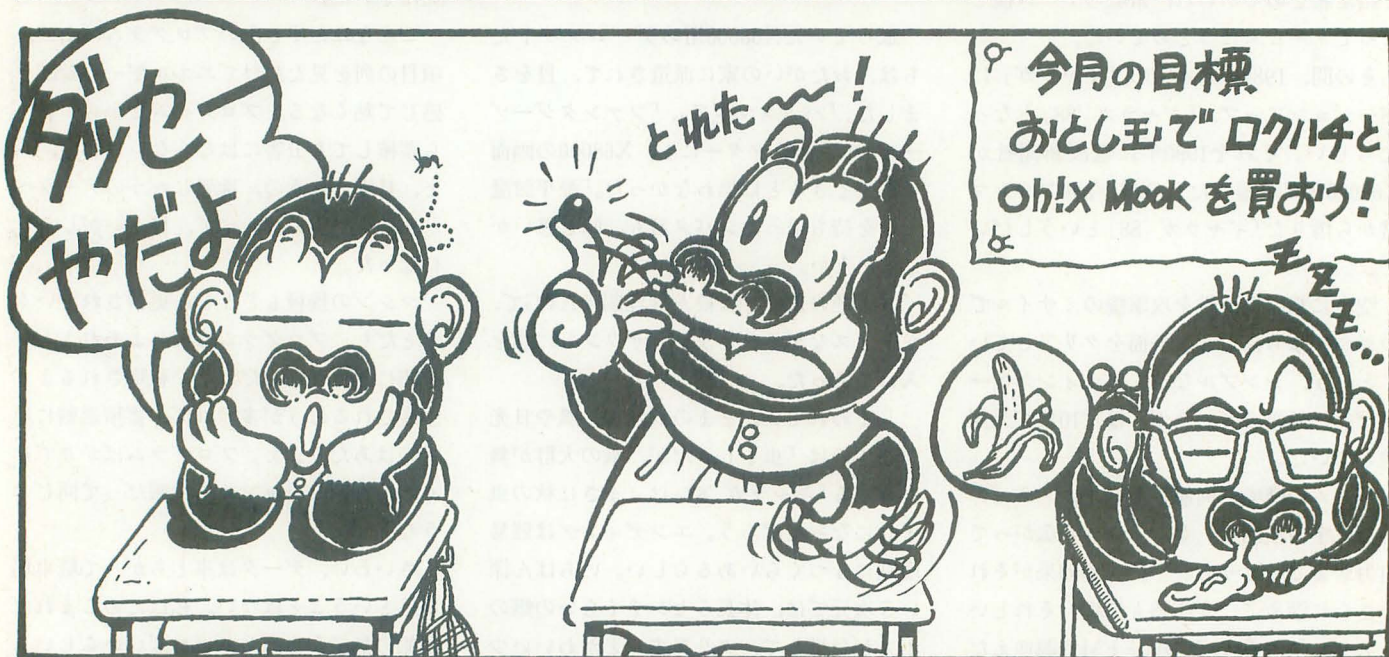
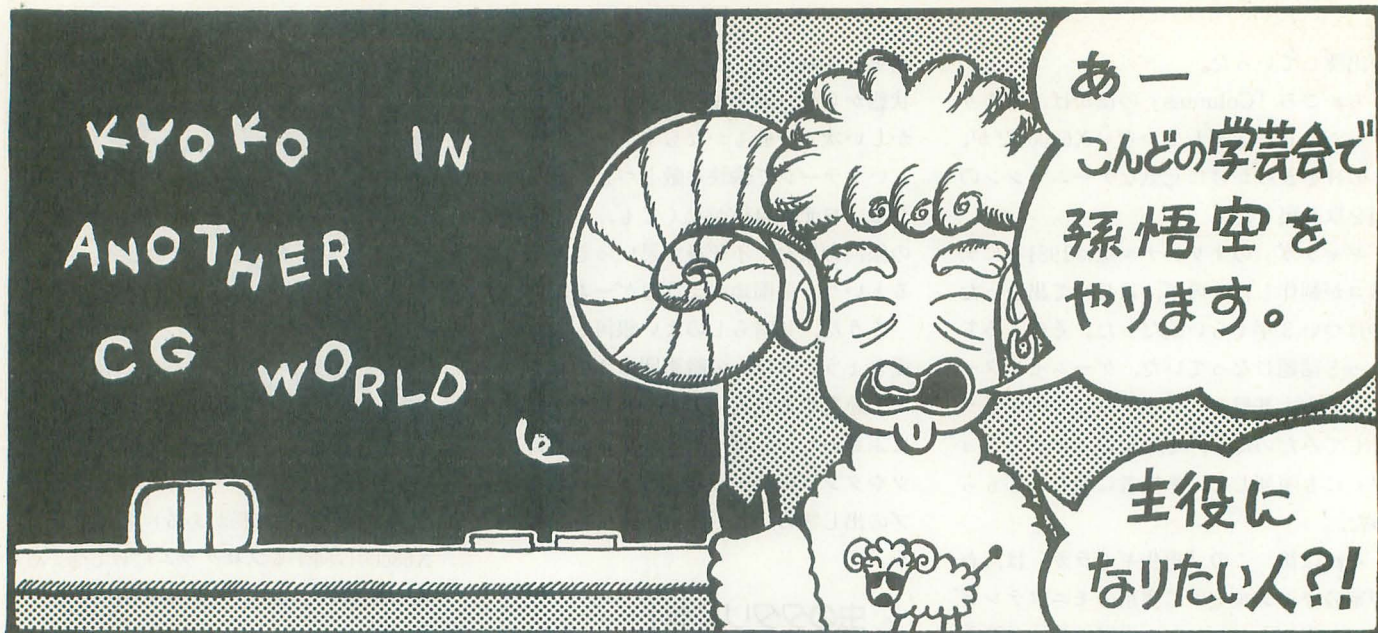
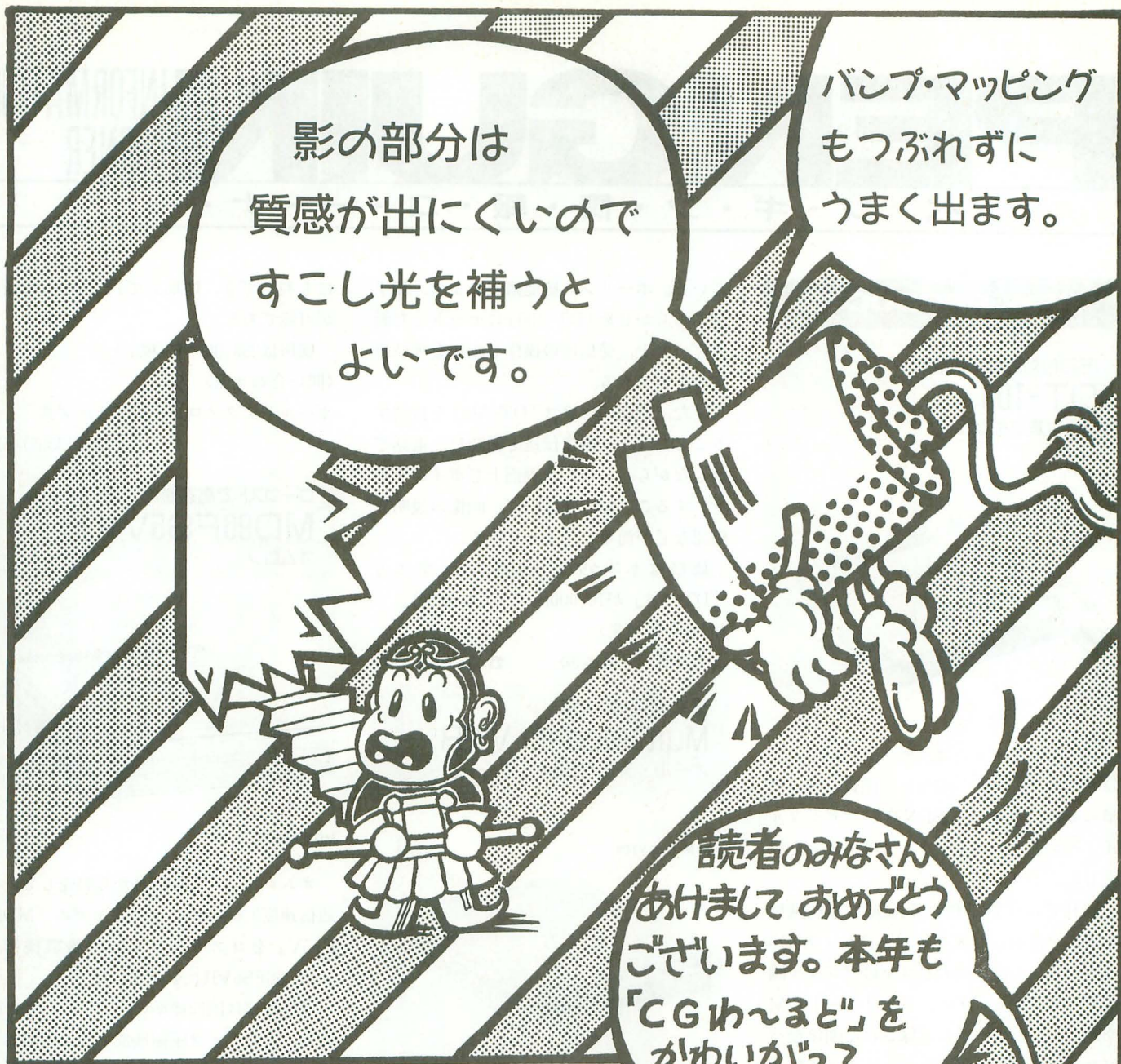
さいわい，データは車とちがって駐車場不足ということはない。私はためこまれた情報の熱気を妄想していればいいらしい。

意外な蓄積が，現実に発見されたのはそのあとだった。長く無人をつづけていたＳ市の家の床下に，事件は発覚した。

木目の床の上に張られた，装飾をほどこしたビニールタイルの一部が，不自然にへこんでいるのに気がついた。カッターナイフを使って思いきって切り開いて見たら，あらわれたのは木の床ではなく，びっしりとつまった，シロアリだった。

一瞬思ったのは，こなしきれないプログラムの隊列だったけれど，これはやっぱり「ギャラガ」のたたりだ！

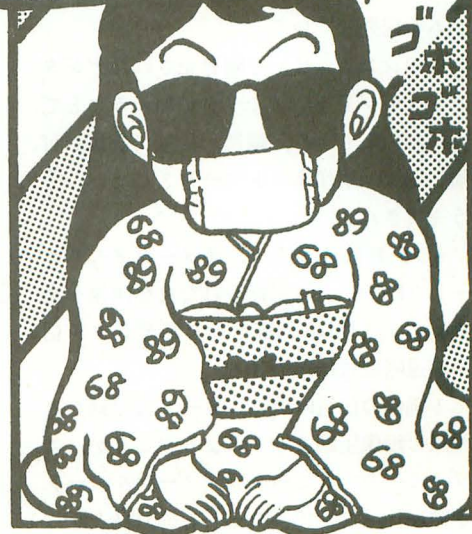
たくわえられて眠っているのが死蔵品なら，蓄積されながら活動をつづけているのは，活蔵品というのだろうか。

「シロアリ討魔伝」は，また後日。

KYOKO IN ANOTHER CG WORLD ⑥
あー
こんどの学芸会で
孫悟空を
やります。
主役に
なりたい人?!

今月の目標
おとし玉でロクハチと
Oh!X MOOKを買おう!
とれた~~!
ダッセー
やだよ
ZZZ...

あっ
そう…
しょせん
猿は猿
とまたしても
思い知らされた
羊先生であった。
目がうつろだよー

今回のCGデータ

総物体数　760
光源　4
1280×1024ピクセル
1670万色フルカラーを4×5ポジで出力
使用ソフトは，C-TRACE，サイクロン
透明体は計算時間がかかるので
必要なだけ使って，効果的に見せる
ようにくふうします。

# PENGUIN INFORMATION CORNER

## ペ・ン・ギ・ン・情・報・コ・ー・ナ・ー

## NEW PRODUCTS

ISDN対応画像伝送機
### DT-100
富士写真フイルム

DT-100

富士写真フイルムは，デジタル信号で記録されたフルカラー画像を高画質のまま簡単に高速伝送する，ISDN対応のデジタルイメージトランスミッタ，「DT-100」を発売した。

同社では世界で初めてADCT方式画像圧縮ICを搭載したメモリカードカメラ「DS-100」と，1枚で最大21コマのデジタル画像が記録できるイメージメモリカード「IM-8S」，およびカードに記録された画像をパソコン，ワークステーションに取り込むメモリカードプロセッサ「DP-100」で構成するフジックス・デジタルスチルカメラシステムを発売している。

このシステムに，今回発売されたデジタルイメージトランスミッタが加わることで，自然画の取り込み，編集，加工に伝送が可能になり，映像ネットワークとしていっそう充実される。

「DT-100」を使うと，イメージメモリカード「IM-8S」に記録された自然画像を高画質のままISDN回線を通じてデジタル信号で遠隔地に伝送できる。

1画像の伝送時間は，カードに記録された際の画像圧縮モードにより異なるが，エコノミーモードで約8秒，ノーマルモードで約16秒，ファインモードで約31秒となっている。ポーリング機能搭載により，相手側であらかじめ「DT-100」にセットした画像データを，受信側の操作で画像を呼び出すこともできる。

また，別売のマウス「DT-M1」を接続することにより，画像伝送と同時に，電話で話しながら，転送した画面上でポインティングすることができるので，画像の説明，確認などが簡単にすませられる。

価格は本体が800,000円で，マウスの「DT-M1」が10,000円（ともに税別）。

〈問い合わせ先〉
富士写真フイルム㈱　☎03(3406)2111

超高速57,600bps
### Multi Modem V32B
オール・テクノロジー・グループ

Multi Modem V32B

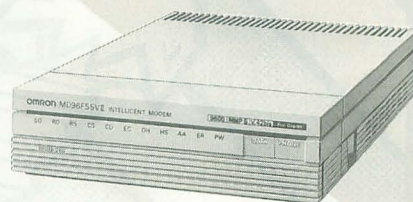
オール・テクノロジー・グループは，「Multi Modem」シリーズの新機種，「Multi Modem V32B」を発売した。

「Multi Modem V32B」はCCITT（国際電信電話）通信規格「V.32bis/14,400bps」を搭載している。また，データ圧縮方式はCCITT V.42bisを採用し，最高57,600bpsのパフォーマンスを得ることができる。CCITT規格V.32bis機能を搭載したモデムの本格的な販売は日本では初めてのこととなる。

一般回線用モデムとしては，V.42エラー訂正，セキュリティ・コールバック，リモート・コンフィギュレーション，UUCP，オートマチック・フォールバックなどの機能を装備している。また，2線式専用回線にも利用でき，同期式で14,400bpsの通信が可能である。

価格は298,000円（税別）。

〈問い合わせ先〉
オール・テクノロジー・グループ㈱
☎03(3503)4516

ローコストで高速通信
### MD96FS5VII
オムロン

MD96FS5VII

オムロンは，昨年12月から発売している通信速度9,600bpsの全2重モデム「MD96FS5V」をコストダウンした後継機種，「MD96FS5VII」を発売した。

通信規格は国際標準のCCITT V.32に準拠し，またデータ圧縮機能として国際標準のCCITT V.42bisとMNPクラス5を搭載している。以上のデータ圧縮機能の搭載によって，ソフトの工夫なしで実効通信速度が最高3倍の約28,800bpsまで向上することになる。

さらに，パソコン本体とモデム間の通信速度を従来の2倍の38,400bpsまで向上させたため，CCITT V.42bisのデータ圧縮機能による実効通信速度28,800bpsがそのまま実現できる。

また，エラー訂正機能として，MNPクラス4とCCITT V.42を標準搭載しているので，ほとんどエラーのない通信が可能になっている。

価格は148,000円（税別）。

〈問い合わせ先〉
オムロン㈱　☎03(5488)3216

無停電電源装置
## BU351X
オムロン

BU351X

オムロンは，薄型の無停電電源装置「BU351XL」のシリーズ追加機種，「BU351X」を発売した。

「BU351X」は価格，機能は「BU351XL」と同等のまま，パソコンの横に置いて使用できるように，縦型のボックスタイプにしたものである。

出力容量は350VAで，標準的なパソコンシステム1台を5分間バックアップできる。幅100mm×高さ190mm×奥行き340mm，重量6.5kgで，縦に置くことも横に置くこともできる。ほかの機能としては，商用電源ラインからのサージとノイズを除去する機能を搭載している。接続する機器から商用電源ラインに流出するノイズも除去する。また，停電，過負荷，バッテリーローの3種類のアラーム機能の搭載で，停電時のバックアップ状況やバッテリー放電状態が容易に確認できる。

価格は59,800円（税別）。

〈問い合わせ先〉

オムロン㈱　☎03(5488)3216

ハンディ液晶プロジェクタ
## HP-40Hi
富士写真フイルム

富士写真フイルムは，今年3月から発売しているハンディプロジェクタ「HP-40」の姉妹機として，小型（幅97mm×高さ62mm×奥行き179mm），軽量（本体重量約430g）を維持しながら，さらに画質を向上させた「HP-40Hi」を発売した。

「HP-40Hi」では液晶パネルの画素数が

HP-40Hi

従来の約55,000から約89,000にアップし，より高い解像度を実現している。

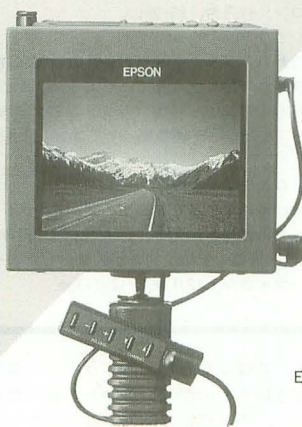
一般のテレビの6型から40型以上に相当する大画面までフリーサイズで映写でき，しかもステレオスピーカーを内蔵しているので，さまざまなシチュエーションでの使用が考えられる。

ACアダプタを使用せずに，8ミリビデオカメラ用の充電式バッテリー(NP-55，NP-77Hなど)で映写することもできる。電池寿命は「NP-77H」を使用した場合，連続投影時間は約30分となる。

価格は59,000円（税別）。

〈問い合わせ先〉

富士写真フイルム㈱　☎03(3406)2111

車載用キット標準装備の液晶テレビ
## ET-562R
セイコーエプソン

ET-562R

セイコーエプソンは，車載用キットを標準装備した液晶カラーテレビ「カーAVシステム　F1」シリーズの最上位機種として，画面サイズ5.3インチの「ET-562R」を発売した。

「ET-562R」は画面サイズ5.3インチの大型，高画質MIMアクティブマトリクス方式液晶パネル（15万画素）を採用している。TVスタンドやステッカー，アンテナなどの車載取り付けに必要な部品もすべて車載用キットとして標準装備。さらに，屋内でもそのまま楽しめるように，デスクトップスタンドと専用ACアダプタも装備した，「デュアルキット」タイプとなっている。

ほかには，カーステレオでも音声を聞くことができる「FMトランスミッタ機能」，電波が途切れると耳障りなノイズを自動的に消音してくれる「音声ミュート機能」などを搭載している。

価格は122,000円（税別）。

〈問い合わせ先〉

エプソン販売㈱　☎03(3377)7001

# INFORMATION

NTTPCネット主催
## 第6回CGコンテスト

第6回を迎えるNTTPCネット主催のCGコンテストが開催される。前回は応募作品数も60となり，同CGコンテスト史上最高の盛り上げを見せた模様。今回からPC-VANも参加するので，6ネットワークでの絵の競演となる。まず，所属ネットでミニコンテストを行い，それぞれのネットで選ばれた作品で順位が決定される。部門は16色以下のみということになり，また，多くの人が参加できるように，初心者，上級者というクラス制がとられている。

詳しくは各ネットの掲示板案内で。

第1回
## Z'sGRACON
ツァイト

ツァイトは，「Z'sSTAFF PRO-68K」，「Z'sTRIPHONY DIGTIALCRAFT」を使用したビジュアル作品を対象に，「第1回ジーズ・グラフィック・コンテスト」(Z'sGRACON)を開催する。応募部門は「年賀状部門」「似顔絵部門」「乗り物部門」「自由課題部門」の4つが用意されている。プロの参加もOK。応募は，所定の応募用紙に必要事項を記入のうえ，フロッピーもしくは出力したもの(サイズは自由)に添付し，郵送する。締め切りは1992年2月29日。

〈問い合わせ先〉

㈱ツァイト　☎03(3299)0461

# FILES Oh!X

このインデックスは，タイトル，注記——筆者名，誌名，月号，ページで構成されています。あけましておめでとうございます。受験生の方，いよいよラストスパートですね。気を引き締めて頑張ってください。

## 一般

▶ぼくらアダルトゲーム大好き!?
脱衣麻雀からHRPGまで。影に隠れたベストセラー，アダルトゲームをさまざまな角度から切りまくる。——編集部，LOGIN，21号，211-225pp.

▶CD-Iってナーニ？（前編）
CD-ROMのようでちょっと違う。絵が出て音が出てインタラクティブに使えちゃうCD-Iっていったい何者？今回はCD-Iの現状と問題提起編。——編集部，LOGIN，22号，238-241pp.

▶ど～するど～なるパソコンゲーム！ PC-9801シリーズの行方
パソコンといえばPC-9801というご時勢だが，さてこのPC-9801とはどういう進化の過程を経てきたのであろうか？ PC-9801シリーズの歴史を検証。——編集部，テクノポリス，12月号，143-146pp.

▶日本パソコン百景
アミューズメントマシンショウを取材しに出かける。背広，ジーンズ，ハイレグにワンレンの超美人とさまざまな人がいきかう会場には，ゲームマシンだけでなくコイン落としやらゲーム機に張るシールやらジェットコースターのビデオプレゼンテーションなどもあって遊戯施設の多様さを感じさせてくれる。——フデヨシ＆カワラ，ASCII，12月号，278-279pp.

▶パソコンで体験する天文学・宇宙の旅
SS433という天体では，光速の26%という速度でプラズマのガスが吹き出す「宇宙ジェット」という現象が観測される。その見え方をシミュレートする。もうひとつは温度による星の色を調べるスペクトル分類。——福江純・岡田里佳，ASCII，12月号，355-360pp.

▶Comdex/Fall
毎年秋にアメリカ・ロサンゼルスで行われる最大のコンピュータ展示会，Comdexの模様をリポートする。マルチメディア，ネットワーク，Windows関連が盛況だったようで，その製品と可能性についても触れる。——高橋三雄，マイコン，12月号，96-99pp.

▶データショウ'91
晴海で行われたデータショウ。各社の新製品群がいっせいに発表されて盛り上がっていたようだ。周辺機器メーカーからも大容量HD，小型プリンタなどが続々。その紹介を行っている。——編集部，マイコン，12月号，106-111pp.

▶エレクトロニクスショウ'91
こちらは10月の1日から5日まで開かれたエレクトロニクスショウの模様を伝える。いよいよ実用段階に入ったCD-I，カラー液晶などのブースが人気だったようだ。——編集部，マイコン，12月号，112-113pp.

▶MYCOM Let's TAKE THE NEXT ONE
年賀状やクリスマスカードを作るのに必需品のプリンタを扱う。いま買える10万円以下のプリンタを一挙に紹介。——編集部，マイコン，12月号，142-145pp.

▶ぱそのこん
パソコンを使ううえで生じる素朴な疑問を解明しようというページ。まず最初の今回は電源を入れるまでのトラブルについて。——島川言成，マイコン，12月号，258-259pp.

▶ビジネスマンのための情報管理術
パソコンと電子手帳DB-Zとのデータ交換ソフト，Hyper HAL-CATCHを取り上げ，データ転送の方法やデータ管理の方法について解説する。——塚田洋一，マイコン，12月号，260-263pp.

▶入門ハード工作室
クリスマスということでミニクリスマスツリーを作る。CDSとメロディICを使って暗くなるとオルゴールが鳴るようになっているぞ。——石川至知，マイコン，12月号，295-299pp.

## MZシリーズ

**MZ-1500(BASIC MZ-5Z001)**

▶平面ルービック
ルービックキューブの2次元版？ タイル並べパズルゲーム。——白井建夫，マイコンBASIC Magazine，12月号，121-122pp.

**MZ-2500(BASIC-M25)**

▶オレは右ききだ！
右腕の腕力には自信があるけど左腕はだめな主人公。ボートを漕ぐと左へ行ってしまう。さあ，どうやって優勝をかっさらう？——謎のパズル大好きおじさん，マイコンBASIC Magazine，12月号，123-125pp.

▶LET'S PROGRAM
今月の課題はN進数の割り算。BASIC-M25を使った読者からの解答例が紹介されている。2進数から16進数までの計算ができる。——藤本健，マイコン，12月号，242-249pp.

## X1/turbo/Z

**X1シリーズ**

▶THUNDERBOLT FIGHTER
戦略性を高めた2人用カード式格闘ゲーム。——らぶた，マイコンBASIC Magazine，12月号，151-153pp.

**X1＋FM音源ボード（要NEW FM音源ドライバ）**

▶LAYDOCK 2 ～オープニング～
T&E SOFTのミュージックプログラム。「レイドック2」オープニングテーマ。——森吉史，マイコンBASIC


参考文献
I/O 工学社
ASCII アスキー
コンプティーク 角川書店
テクノポリス 徳間書店
POPCOM 小学館
マイコン 電波新聞社
マイコンBASIC Magazine 電波新聞社
LOGIN アスキー


## 新刊書案内

ああ，昔はよかった。何しろ，パソコンゲームにはどういうものが向いているかがわかっていなかったし，売れ筋に支配されることもなかった。良識などどこにもなく，デタラメで，クセのある変なモノが，まっとうなゲームに混じって出回っていた。

ボコスカWARS（なんていいタイトルだ！），EMMY II（CRTの中の女の子とキーボードで対話するバカなゲームだ），テグザー（まぎれもない傑作！），ブラスティー（アニメすればいいってもんじゃない），プロデュース（恨みを持つ人物を妖怪を使って驚かせるという，ものすごいコンセプトのゲーム）などなどである。きわめつけは「177」。こいつはスゴいゲームだった。結局，発売は途中で中止になったが，家路に着く女の子を追いかけて××するというシャレにならないゲーム。177というのは「刑法177条」のこと。恐ろしい時代だ。

本書は，とにかく強烈なゲームを拾いながらパソコンゲームの歴史を紹介するという本である。全部ひとりで書いているところが残念だ。今度は私も呼んでおくれ。（K）

**パソコンゲームの達人　伊藤哲郎著　秀和システムトレーディング刊　A5版　266ページ　1,400円**

書籍の価格は消費税込みです

Magazine，12月号，181-182pp.

X1turboシリーズ

▶竜騎士伝説

真の竜騎士になるために試練の山へ向かえ！ アクションRPG。——浪越孝宏，マイコンBASIC Magazine，12月号，154-155pp.

## X68000

▶NEW SOFT

メルヘンチックアクション「NIKO²」を紹介。——編集部，LOGIN，21号，21p.

▶最新ゲーム徹底解剖!!

前号に引き続き，アクアレスの攻略。——編集部，LOGIN，21号，160-163pp.

▶SOFTWARE REVIEW

家庭用ゲームマシンの性能が飛躍的に向上し，X68000の地位に陰りが？ しかし，まだまだ負けてはいられない！ キャメルトライでX68000ゲームソフトを検証。——三宅ボル助，LOGIN，21号，172-173pp.

▶X68000新聞

プロサッカー68，キャメルトライ，全開電飾，麻雀マスター，出たな！ツインビーを紹介。札幌にある，X68000ゲーマーなら誰でも知っているソフトハウス，ズームを訪問。——編集部，LOGIN，21号，236-241pp.

▶NEW SOFT

プロサッカー68，ラストバタリオンを紹介。——編集部，LOGIN，22号，23-24pp.

▶最新ゲーム徹底解剖!!

パワーモンガーの攻略。——編集部，LOGIN，22号，150-155pp.

▶X68000新聞

あの鳥居部長がX68000部隊へ帰ってきた！ X68000の生みの親，鳥居勉さんの電話インタビュー。新作ソフトは，ジェノサイド2，ブリッツクリーク，サイバーコア，F15ストライクイーグルII，ディノランドの紹介。第1回X68000芸術祭の模様と入賞作品紹介。——編集部，LOGIN，22号，256-261pp.

▶GAMING WORLD

不思議なアクションパズル，レミングスの攻略ほか，プロサッカー68，ボナンザブラザーズ，ラストバタリオン，飛翔鮫，サイバーコアの紹介。新作ゲームの先取りは，出たな!! ツインビー，ジェノサイド2，CODE-ZERO，ディノランドなど。——編集部，テクノポリス，12月号，20-26，33-40pp.

▶SOFT EXPRESS

ノーブルマインド，麻雀マスター，キャメルトライ，ジェノサイド2，ラストバタリオン，プロサッカー68，サイバーコア，全開電飾の紹介。——編集部，コンプティーク，12月号，108-111pp.

▶HOW TO WIN

「パワーモンガー」の攻略と，開発中のアリシミュレータ「シムアント」を紹介。——編集部，コンプティーク，12月号，186-189pp.

▶Software Hot Press

レミングス，サイバーコア，ジェノサイド2を紹介。——編集部，POPCOM，12月号，18-30pp.

▶マッド寿のこだわりレポート 銀河を我が手に！

銀河英雄伝説IIDX＋Kitの紹介と攻略。——マッド寿，POPCOM，12月号，70-71pp.

▶ゲームの達人

“力”のゲーム「パワーモンガー」の攻略。——編集部，POPCOM，12月号，108-109pp.

▶誌上公開質問状

X68000専用TVチューナー「CZ-6TU」の仕様，日本語入力時の環境ファイルについての質問に答えている。——多田太郎，マイコンBASIC Magazine，12月号，91p.

▶林檎の木

ブーメランで林檎を落として取ろう。限られた回数ですべての林檎を取れば面クリア。——船瀬竜久，マイコンBASIC Magazine，12月号，156-157pp.

▶ARMY

ザコキャラいっぱい，でかキャラ3匹倒す。ジャンプアクションゲーム。——遠藤克之，マイコンBASIC Magazine，12月号，158-159pp.

▶タコタコウォーズPART 2

新たなる戦いがいま始まる。パンチやキック，ヘディングやジャンプで相手のLIFEを早くなくして倒せば勝ち。——高橋秀之，マイコンBASIC Magazine，12月号，160-161pp.

▶Phantasy StarII ～ライズ オア フォール～

セガのゲームミュージックプログラム。要NAGDRV＋MT-32系MIDI楽器。——少年一号，マイコンBASIC Magazine，12月号，183-184pp.

▶ウイニングラン

ナムコのゲームミュージックプログラム。要NAGDRV＋MT-32系MIDI楽器。——渡辺祐介，マイコンBASIC Magazine，12月号，185-187pp.

▶今月の注目ソフト スターウォーズ

ワイヤーフレームがいかにもそれらしい3Dシューティングゲーム「スターウォーズ」を紹介＆攻略。各ステージの解説。パワーモンガーの攻略法。——山下章・解せないクン，マイコンBASIC Magazine，12月号，238-241pp.

▶AV STRASSE

SX-WINDOW対応のグラフィックデータ集第1弾「SX-WINDOWイラスト集VOL.1 一般実用編」と，ボナンザブラザーズ，キャメルトライのゲーム2作を紹介。——編集部，ASCII，12月号，369-372pp.

▶TBN GAME

イマジニアから発売された「パワーモンガー」を扱う。ゲームのシステム，攻略法などを紹介。——編集部，ASCII，12月号，402-404pp.

▶FREE SOFTWARE INDEX

今月のPDSアップロード情報。ディスクドライバ，コンソールドライバなどX68000の新参フリーウェアが掲載されている。——編集部，ASCII，12月号，447-451pp.

▶HOBBY EXPRESS

AMショウのレポートと，プロサッカー68，パワーモンガーのレビューを掲載。——編集部，マイコン，12月号，315-339pp.

▶なんでもQ&A

Multiwordのグラフィック拡大方法，ウィンドウモードでの範囲先指定はテキストモードでは使えないか？ などの質問に答える。——シャープ株式会社電子機器事業本部AVCシステム事業推進室，マイコン，12月号，390-391pp.

▶GAME BOX

イマジニアのプロサッカー68を取り上げる。派手ではないが本格的なサッカーゲームだ。——トマム・ニセコ，I/O，12月号，115p.

▶RED BALL

2人対戦型のシューティングゲーム。漂うBALLをよけながらビームを相手に打ち込むのだ。——沢田広正，I/O，12月号，177-179pp.

▶Run Away

刑務所から脱走中の「ゴン太」を操って，スクロールしてくる刑務所の壁の隙間を通って脱出をはかるというゲーム。——野崎広之，I/O，12月号，180-181pp.

## ポケコン

PC-E500

▶ブロックくずし1991

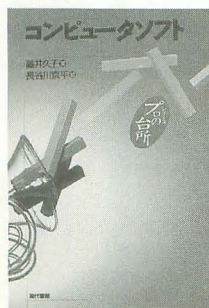
なんとボールもブロックも1ドット。パワーアップ型ブロックくずしゲーム。——森内俊爾，マイコンBASIC Magazine，12月号，164-165pp.

▶ENONE 2

動き出したら止まらない！ パズルアクションゲーム。——佐藤祐紀，マイコンBASIC Magazine，12月号，166-167pp.

PC-1600K

▶PC-1600K実践プログラミング

シャープのポケコンPC-1600Kを例として，ポケコンをパーソナルレベルで活用する方法を解説する。今回はAREAD命令を使って，前回解説した定義づけプログラムをさらに発展させる。——塚田洋一，マイコン，12月号，264-266pp.

**コンピュータソフト**

『パソコンからファミコンまで，コンピュータと呼ばれるものは数々あれど，ソフトがなければただの“ハコ”』。機械にうとい筆者が，ソフトウェアに興味を持ち，ひとつのソフトができるまでを取材しまとめたもの。ゲームやビジネスなどさまざまな分野のソフトハウスを訪れ，ソフト制作における喜怒哀楽や仕事の具体的な内容などを，現場の人間に直接取材している。もちろんウラ話なんかもちらほらあったりする。

**藤井久子著 現代書館刊 ☎03(3261)0778 新書判 206ページ 1,380円**

**GAME ゲーム げぃむ**

幻夢年代記の続編で，ログイン誌に連載されていたコラムをまとめたもの。前作は主にアメリカのゲームを中心に話を進めてきたが，それから2年たった今回は，ヨーロッパのゲームを主軸に書かれている。筆者のさまざまなコンピュータゲーム体験をとおして，たった2年ではあるがゲームのあり方が確実に変わってきているのがわかる。数々のゲームが紹介されているので，読み物としても存分にたのしめる1冊だ。

**安田均著 アスキー出版局刊 ☎03(3486)1977 新書判 319ページ 1,800円**

書籍の価格は消費税込みです

# QUESTION and ANSWER

最近，マシン語をやり始めたのですが，よくわからないことがあったので質問します。このあいだ，メモリーメモリ間転送プログラムを作ろうとして，次のようなプログラムを作ってみました。

```
      LD  HL,8000H ;転送元アドレス
      LD  DE,9000H ;転送先アドレス
      LD  BC,400H  ;転送サイズ
LOOP:
      LD  A,(HL)
      LD  (DE),A
      INC HL
      INC DE
      DEC BC       ;カウンタを減らす
      JR  NZ,LOOP  ;ＢＣが０でなければ
繰り返す
      RET
```

このプログラムを実行すると，無限ループになってしまうのです。あとになって，これと同じ動作をするLDIR命令を知ったのでとりあえずプログラムは作れましたが，なぜ無限ループになってしまうのか，いまだにわかりません。初心者ということでよろしくご回答願います。それから使用機種はＸ1turboです。　群馬県　今井　良一

このような繰り返し構造のプログラムを作る場合，繰り返す回数が256回以内ならＢレジスタをループカウンタにしてDJNZ命令が使えるので便利なのですが，質問のように繰り返し回数が1024回では，ループカウンタにレジスタペアを使わざるをえません。実は，ループカウンタにレジスタペアを使った場合は思わぬ落とし穴があるのです。Ｚ80の上級者でも一度は同じようなバグを経験したことがあると思います。しかし，ここでは結論を話して終わりにするのではなく，なぜこのプログラムがうまく動かないのかちょっと考えてみることにしましょう。

まず，無限ループに陥る原因を考えてみましょう。

1)　ループカウンタ
2)　ループの終了判定

このどちらかが間違っていると思われます。

ループカウンタがおかしい例としては，

```
      LD   B,20
LOOP:
      LD   B,(HL)
      ADD  A,B
      DJNZ LOOP
      RET
```

のようなものが考えられます。これは，ループの中でループカウンタであるＢレジスタを破壊してしまっているために，正しい繰り返し回数が実行されないものです。

さすがに，これだけ単純なプログラムでループカウンタを破壊するようなバカはいないでしょうが，プログラムが複雑になればうっかりミスもでてきます。

そこで質問のプログラムを検証してみると，ループカウンタはBCレジスタであり，１回メモリ内容を転送するたびにBCレジスタを１減らしていますね。ループカウンタであるBCレジスタは，DEC BCで減算される以外操作されることはないので，おかしいところはないようです。ということは残された可能性として，ループの終了判定があやしくなってきますね。

ループの終了条件はゼロフラグが立つまで（1にセットされるまで）になっています。つまり，「DEC BCでBCレジスタが０になったところで，ゼロフラグが立つはずだから，ゼロフラグが寝ているあいだはループを続けさせよう」と今井さんは考えたんだと思います。この考えは一見正しいようですが大きな間違いがあるのです。

結論をいいますと，実はDEC（INC）命令は操作対象が16ビットの場合，演算結果がフラグに影響しないのです。つまり，Ｂレジスタが１で，

　DEC B

は，ゼロフラグを１にセットしますが，BCレジスタが１で，

　DEC BC

は，ゼロフラグになんの影響も与えないのです。質問のプログラムを見ると，フラグに影響を与える命令がひとつもありません。そのために，終了条件であるゼロフラグが

リスト1

```
==================  gramuse.s  ==================
     1: *
     2: * GRAMの使用の有無を
     3: *   環境変数 'gram' に返すプログラム
     4: *
     5: * gram = use      GRAMは使われています
     6: *      = no_use   GRAMは使われていません
     7: *
     8: _TGUSEMD:         equu     $e
     9: _EXIT:            equ      $ff00
    10: _SETENV:          equ      $ff52
    11:
    12:         .text
    13:
    14:         moveq.l #_TGUSEMD,d0     * IOCSコール TGUSEMD
    15:         moveq.l #0,d1            * GRAMの
    16:         moveq.l #-1,d2           *   使用状況を
    17:         trap    #15              *   レポート
    18:
    19:         tst.l   d0               * 戻り値が
    20:         beq     no_use           *   0なら使っていない
    21:         cmpi.w  #3,d0            * 3以外なら
    22:         bne     use              *   システム／アプリケーション
    23:                                  *   で使われている
    24:
    25:         moveq.l #_TGUSEMD,d0     * 使用した後、壊れたままの
    26:         moveq.l #0,d1            *   GRAMを
    27:         moveq.l #0,d2            *   未使用に
    28:         trap    #15              *   設定する
    29:
    30:         * 嫌なら上の4行を無効にすること
    31:
    32: no_use:
    33:         pea.l   no_use_mes       * 未使用のメッセージ
    34:         bra     set_env
    35: use:
    36:         pea.l   use_mes          * 使用中のメッセージ
    37: set_env:
    38:         clr.l   -(sp)            * 自分の環境を指定
    39:         pea.l   env_name         * 環境変数名"gram"に
    40:         dc.w    _SETENV          *    メッセージを設定
    41:         lea.l   12(sp),sp        * スタック補整
    42:         dc.w    _EXIT            * プログラム終了
    43:
    44:         .data
    45:
    46: env_name:
    47:         dc.b    'gram',0         * 環境変数
    48: use_mes:
    49:         dc.b    'use',0          * 使用中
    50: no_use_mes:
    51:         dc.b    'no_use',0       * 未使用
    52:
    53:         .end
```

リスト2

```
==================  wwp.bat  ==================
     1: echo off
     2: gramuse
     3: if %gram% == use goto abort
     4: wp
     5: goto end
     6: :abort
     7: echo GRAMは使われています
     8: :end
     9: echo on
```

リスト3

```
 10 /*
 20 /*          WP.X書き換えプログラム
 30 /*
 40 /*              BY H.K/Z.N
 50 /*
 60 /*
 70 int         ai,bi,ii
 80 str         dummy,ver,check="1.10"
 90 /*
100 cls
110 print"0. 無変換
120 print"1. 一括変換
130 print"2. 一括変換&辞書先読み
140 print"3. 逐次変換
150 print"どれを選びますか?";
160 repeat
170 dummy=inkey$
180 until dummy>="0" and dummy<="3"
190 print dummy
200 bi=asc(dummy)-48
210 /*
220 wait()
230 /*
240 ai=fopen("a:¥wp¥wp.x","rw")
241 ii=&H15F2F6-&H130000
250 fseek(ai,ii,0)
260 freads(ver,ai)
270 if ver<>check then print "バージョンが違います。":end
280 ii=&H13A18C-&H130000+3
281 fseek(ai,ii,0)
290 fputc(bi,ai)
300 fclose(ai)
310 print "終了しました。"
320 end
330 /*
340 func wait()
350 print "ドライブ1にWP.Xの入ったディスクを挿入して下さ
い。"
360 print "準備が出来たら何かキーを押して下さい。"
370 dummy=inkey$
380 endfunc
```

立つことは永遠になく，無限ループになってしまうわけです。

このようにレジスタペアをDEC (INC)したときの結果が0であるか調べたいときは，Aレジスタを介して，

DEC BC
LD A,B
OR C

のように,上位と下位のORを取る方法があります。これで上位，下位とも0のときに，ゼロフラグが立つようになります。

**X68000でG-RAMの領域をRAMディスクとして確保していて，うっかりWP.Xを起動するとRAMディスクが壊されてしまいます。WP.Xの起動前にグラフィックRAMが使用されているか調べて，使われているようなら起動を中止するようなバッチファイルを作成したいのですが，どうすればいいのでしょうか。　愛知県　飯塚　秀治**

テキスト・グラフィックRAMの使用状況を設定，あるいは調べるIOCSコールが，TGUSEMD (IOCSコール番号$0E)です。もしも個人でテキスト・グラフィックRAMを使用するプログラムを作ろうというときは，必ずこのIOCSコールで“グラフィック(テキスト) RAMをアプリケーションで使用する”ことを宣言しておきましょう。実行後は“使用していない”を宣言するのを忘れずに。

勘違いしてもらっては困るのですが，このIOCSコールでG-RAMの使用を宣言したからといって，OSがその後，G-RAMを使用するようなプログラムに対して警告を発するわけではありません。ですから，G-RAMを使用するプログラムを作る場合は，G-RAMの使用の有無をTGUSEMDで調べる必要があります。WP.XがG-RAMを破壊するにもかかわらず，使用チェックをしていないのにどんなわけがあるか知りませんが，メーカーの作成した標準添付のソフトがこれでは呆れてしまいますね。

さて，バッチファイルからG-RAMの使用状況を調べるために作成したプログラムがリスト2です。解説は省きますが，リスト中にたくさんコメントを付けておいたので，それらを参考にしてください。

プログラムを実行すると，環境変数“gram”に，G-RAMが使われていると“use”，使われていないと“no_use”を設定します。しかし先ほども話しましたように，G-RAMを使用していてもTGUSEMDで使用を宣言してない場合は未使用と判断されますから，これが絶対ではありません。

それからリスト1は，G-RAMの使用状況が“アプリケーションで使用したあと壊れたまま”になっていると，勝手にG-RAMの使用状況を“使用していない”に設定するようになっています。というのは世の中にあるG-RAMの使用をチェックしているソフトのいくつかは，TGUSEMDの戻り値が0以外だとG-RAMが使用されていると判断して起動できないものがあるからです。

ちなみに“アプリケーション……壊れたまま”の戻り値は3であり，WP.Xを使ったあとはそのように設定されています(私が持っているものでは，フリーウェアのGRAD.rがWP.Xの使用後にG-RAMをRAMディスク領域にとることができない。まあ，Nスイッチで強制することができるからいいんだけどね)。それが大きなお世話だという人は，リスト中のコメントを参考にして，対応する行を削除してください。

最後にリスト2は，WP.Xの起動を例にとったバッチファイルの作成例です。参考にしてください。　(影山　裕昭)

**X68000本体付属ワープロのバージョン1.10で起動時から「一括変換」を使用したいのですがどのようにすればよいのでしょうか。以前，古いバージョンについては解説があったようなのですが。　新潟県　安田　滋富**

バージョン1.01のほうはOh!X1990年の2月号の質問コーナーで影山氏がサンプルを示していますね。では1.10のほうのパッチプログラムをリスト3に示します。起動後指示に従って操作してください。実行の際は必ずバックアップしたものを使い各自の責任の下で行ってください。(西川善司)

**質問にお答えします**

日ごろ疑問に思っていること，どんなことでも結構です。どんどんお便りください。難問，奇問，編集室が総力を挙げてお答えいたします。ただし，お寄せいただいているものの中には，マニュアルを読めばすぐに回答が得られるようなものも多々あります。最低限，マニュアルは熟読しておきましょう。質問はなるべく具体的に機種名，システム構成，必要なら図も入れてこと細かに書いてください。また,返信用切手同封の質問をよく受けますが，原則として，質問には本誌上でお答えすることになっていますのでご了承ください。なお，質問の内容について，直接問い合わせることもありますので,電話番号も明記してくださいね。

**宛先：〒108 東京都港区高輪2-19-13**
**NS高輪ビル**
**ソフトバンク株式会社出版部**
**「Oh! X質問箱」係**

## FROM READERS TO THE EDITOR

慌ただしい師走が走り去ったあとにやってくるお正月。重そうに着飾った女性たちを見て，今日はなんかのパレードか？などとボケをかまさず，初詣でぐらいは行きましょう。でも，正月はやっぱり寝正月がいちばんだね。

◆X68000での3Dゲームは，まだまだ未開拓の領域にあるといえるでしょう。こう考えた理由のひとつは，ゲーム自体の問題として360°全方向から迫りくる敵とか，自動追尾ミサイルなどがあったら難しすぎてゲームにならないはずです。かといって「アフターバーナー」のようなタイプのゲームはやりあきたし。もうひとつはハードウェアの問題です。X68000でポリゴンを使うのはしんどいし，ワイヤーフレームだと複雑な形を表現するのが困難ですから。しかし，「スターウォーズ」には期待してますよ。

奥井　一穂(18)石川県

ゲーム自体は従来のようなものでも「スターウォーズ」のように，トレースプレイや視点切り替えなどの自由度があれば，ゲームは面白くなるでしょうしね。

◆「スターウォーズ」には期待大。たぶん購入すると思います。でも，ステージは映画「スターウォーズI」だけでなく「スターウォーズIII」にあったデス・スター中央原子炉をめぐる戦闘シーンも入れてほしかったです。爆発から間一髪脱出するシーンなんかは盛り上がると思うんですけどね。う～む，でも私はハン（というかハリソンフォード）のファンなので，彼の役にはまって，チューイを隣にM・ファルコンで活躍したいなあ。　鈴川　美佳子(18)東京都

夢がかなうのはいつになるのだろうか。

◆「パワーモンガー」は危険なソフトだ。買って1日目に思いっ切りはまってしまい風邪をひいた。おかげで会社を休んでしまったのだ。「ポピュラス」にもはまったけど，体を壊すほどではなかったしなあ。では，気をつけましょう羊の食べ過ぎ。今夜もマウスが唸る！

高木　宣博(21)大阪府

会社辞めますか，「パワーモンガー」やめますか？

◆今度は「MORTAL」を引っさげて，柴田さんが登場しましたね。それにしても人間のアニメパターンを60種類も作り出すなんて，キャラクタグラフィックのS-OS上では究極に近かったのではないでしょうか。しかし，柴田さんでもアイデアに困ることがあるんですね。本当に，何かいいアイデアがあればいいのですが。

森田　照美(17)山口県

森田さんの虹色の脳細胞に期待しよう。

◆S-OS用ゲーム「MORTAL」はとてもよかった。遊び始めたときにはルールがのみ込めなくて少し戸惑ったけど，いったん把握してしまってからはのめり込んでしまった。それぞれの教徒たちがうごめいているのを見ると，あるひとつの世界を感じてしまうほどだ。教徒が10人もいるときに溶岩がたくさん流れてくると，緊迫した雰囲気を感じさせてくれるし，溶岩が転がってきたときに教徒たちがぎりぎりのところで逃げるところなんか，見ていてハラハラします。画面スクロールも気持ちいいし，面白かったです。

三沢　弘之(20)神奈川県

皆さんに楽しんでもらって柴田氏も喜んでいることでしょう。

◆11月号32ページの写真を見てびっくらこいた。「なぜ，僕の写真が……」要するに突如現れた謎の男というのが，自分でもいうのもなんですけど本当に似ていると思います。見かけたら話しかけてください。　今川　健久(16)愛知県

話しかけるのはいいですが，石を投げたり餌を与えないでください。

◆11月号の「第3回サイクロンCG大会」を見て思いました。「やっぱり自分との技術がまさに雲泥の差だなあ」と。自分がX68000を買ったのは3DCGをやりたかったからなんですよね。でも，いまだにうまく使いこなせなくて。だからCGの特集があればとてもありがたいです。

弥三谷　彰憲(17)石川県

技術なんてやっているうちに自然と身につくものですよ。とりあえず地道に数をこなしていきましょう。

◆3Dの美少女ソフト。想像するだけで胸がワクワクしませんか？　う～ん，はやくどっかから出ないかなあ。サイクロンCG大会やDōGAさんの作品を眺めていると，フッと思うんですけどね。真面目な話，僕ひとりの願望じゃないと思うけど。　加藤　英俊(24)東京都

ゆくゆくはバーチャルリアリティシステム搭載の美少女ソフトですか？　ここまでくるとさすがに馬鹿らしくなってくるな。

◆9月26日キャメルトライのパッケージが家に届いた。な～んか箱が大きいなあ，とわくわくしながら開けてみるとなんとパドル付！　これがホントの「アーケード版完全移植」だなあと，さらに感心しながらパドルを組み立て始めるとさあ困った。ビスがなかなか入らない。結局，苦闘50分間のすえ完成したわけですけどゲーム自体よりパドル作りのほうが難しかった。まさに1粒で2度おいしいソフトだった。

小原　健一(18)宮城県

やるな！　電波新聞社ってとこですね。

◆「キャメルトライ」ゲームレビューのところで「カモノハシ」というキャプションがあるのですが，どこかで「作者はペンギンだといっている」という話を耳にしたけどどうなんでしょう？　僕も「カモノハシ」だと思うのですが。

金沢　輝一(17)青森県

はにゃららばってん，本当は「ペンギン」だったようです。ごめんなさい。

◆現在，マンデルブロ集合を実部0.26～0.30，虚部0.45～0.49，最大計算回数2000で描画中です。固定小数点演算をさせても48時間を経過し

◀岩瀬　貴代美　福岡県
トップバッターはお馴染みの岩瀬さん。さすが常連だけあって，いいねえ，うまいねえ。学祭用に原稿を描いたそうだけど見てみたいなあ。

◀鈴川　美佳子　東京都
ドラッケンといえば，移動モードで高速スクロールビュンビュンにして，ゲームの目的など無視して遊んでいたっけ。

ているいま，計算はまだ終わっていません。X68000を封印したい人は，死ぬほど時間のかかる計算をさせたりするのもひとつの手ではないでしょうか。結果も残りますし。

枝松　樹(21)愛媛県

短気な僕にはちょっと我慢できそうもない方法だと思う。

◆また，愛機が壊れてしまいました。シャープさんに手紙を出したところ無償で修理していただき大変感謝しています。ところで最初の修理のとき，きれいにビニール袋に入れて持って帰ってもらったところ，戻ってきたらビニールがありませんでした。だから２度目の修理のときは，箱にそのまま入れて持って帰ってもらいました。「今度こそは何もなくならないだろう」と思っていたら，なんとプリンタ端子のカバーがなくなっていました。現在,「次に修理に出したときには何がなくなっているのだろうか」と心配（？）になっています。

藤原　彰人(21)岡山県

きっとシャープのサービスステーションには，いたずら好きの小人さんがたくさんいるんですよ。

◆皆さん，ファランクスの７面を不要だといいますが，私は決してそう思いません。確かにワープ面はまだるっこしいのですが，ワープを抜けたあとの演出が泣かせますねえ。遠くのほうで戦闘している明かりが，だんだん近づいてくる様子は，とても好き！　なのだ。

尾松　剛(18)京都府

きたきた。うれしいねえ，僕も１面の次に７面の演出が大好きなんですよ。

◆80Mバイトハードディスクについて，マニュアルには「フロッピー約80枚分の記憶容量」と書いてます。実際は，81×1024÷1221≒67.9となり「約68枚分」と書くのが正しいと思います。JAROにたれこみしてもよろしいでしょうか？

原　政徳(18)愛知県

そんなかたいこといわずに「JAROってなんじゃろ」と，ぼけをかましつつ笑って許してあげましょう。

◆「ヘビーノヴァ」いやあ，いいですね。あのマイクロネットの格闘モノといえばその昔X1に出ていた「チャンピオンプロレス」や「ロボレス2001」を思い出します。しかも，TAKERUで5,600円とくれば買うしかないじゃないですか。しかし,「スターウォーズ」に「飛翔鮫」もほしい。お金がいくらあっても足りないぞ。

近藤　英二(20)愛媛県

こういうときには，困ったときのバイト頼み。決して借金はしないようにね。

◆最近，私のX68000はちょっと怖いことになっています。９月の初旬，1MビットDRAMを24個買ってきてKGB-PBKII（計測技研の増設メモリボード）にハンダ付けしました。そのときちょっとミスしまして，２枚のメモリチップ間とコンデンサが密着する形になってしまったので，力ずくで引き離したのです。それ以来，電源投入時に6Mバイトあったメモリが3，4時間使っていると３Mバイトに減っているという具合になってしまいました。後悔先に立たず，ということわざが身に染みている今日この頃です。ところでいまではメモリにディスクキャッシュするゲームが増えていますね。先日も「ロードス島戦記」をやっていて「バスエラー」を出してしまいました。心当たりがあるだけに悔しがる気にもなれず，ひたすら悲しかった。それにしてもこんなアドレスまでキャッシュに使っていたんですね。ちなみにこのバスエラーの表示は，見慣れたものとは違います。お目にかかる人はほとんどいないでしょう。

渡辺　一十六(26)埼玉県

ちょっと大きな代償ですね。

◀小川　裕美　山口県
コントラストがはっきりして見やすいし，よくまとまっているね。背景の漢文は模試の問題文というのが泣かせるね。

◀寺門　修司　兵庫県
ひとり，ふたり……う〜ん全部で15人のサンタさんがかわいいね。寺門君はサンタさんに何をお願いしたのかな？　そんな年じゃないか。

◆「出たな!!　ツインビー」が出るんですか。このゲームは面白そうだったけど「ベルが取れない」「背景がムゴイ」ってことで１回しかやっていないな。「パロディウスだ！」の背景は敵や弾が見やすいように，控えめの色調でかつ細かく描き込んであってよかったのに。まあ，メローラさんがかわいいからいいや(おいおい)。ところでコナミといえばMSX2の「メタルギア」をX68000用にグレードアップして移植してくれないかなあ。敵に見つからないように隠れながら進むのがなかなかGOODでしたよ。

石田　伯仁(18)神奈川県

最近のコナミのがんばりは評価したいですね。次は何をやってくれるのかな。

◆歯磨き粉をつけて擦ってみたところ，本体とディスプレイについたヤニやくすんだ色がきれいに落ちました。皆さんもだまされたと思ってやってみてください。うれしくなって「あきほちゃん」と命名しようとしたらシステムディスクが壊れた。怖かった。家田　貴之(22)愛媛県

最近の酵素はがんばるねえ。

◆最近，彼女がX1マニアタイプの所有者であることが判明した。私が「うらやましいなあ」というと「私と結婚した人にのみもれなくついてきます」との返事。ちなみにもう彼女もX1も予約済みです。

伊藤　徹(21)千葉県

……はいはい，次のハガキにいきましょうか。はあ。

◆就職も無事決まり，卒業研究も予定よりかなり早く終わり，卒業論文もほいほいっと終わってしまったので現在放心状態です。おかげでX68000に接する時間が長く，AM3：00就寝，PM2：00起床の毎日です。学校に行かなきゃ，と思いながらつい面倒になって登校拒否７日目になってしまった。生活戻さなきゃ。

谷口　博一(25)大阪府

ほらほら，あんまりだらしないと就職してから大変ですよ。

◆久しぶりにカードゲームが掲載されたのでさっそく入力したヨーン。カードゲームを１枚のディスクに入れて，ファンクションキーで起動するようにしてあるけど，まだファンクションキーに空きがある。早くF20まで空室なしにしたいのでこれからも続けて掲載してくださいね。

佐藤　正彦(53)新潟県

12月号ではお休みしちゃったけど今月，来月号と掲載していきますので，楽しみにしていてください。

◆どうもすみません。いやあ，「Oh!PC」の付録にあったフロッピーディスクをもらったんですが，置いといてもしかたないからフォーマットして使わせていただきました。ソフトバンクの皆様，特に「Oh!PC」編集部の方々，この罪深き子羊をお許しください。山田　直貴(20)兵庫県

そのフロッピーディスクも有効に使われて幸せなんじゃないでしょうか。

◆岡村さんの４コママンガが好きです。私はひとりっ子なもので，ほんわかした兄弟（姉妹）というものに少し憧れてしまうのです。しかし，以前このことを姉のいる友人に話したら「それはお前が姉弟の悪い面を知らんからや」というようなことをいわれてしまった。そのときは納得してしまいましたけど。とにかく，岡村さん次回作を楽しみにしてますよ。

河野　暢(18)大阪府

姉弟のどつき漫才はどこまで突っ走るのであろうか。

◆11月号の電脳倶楽部の広告の写真を見て感激してしまいました。僕を安藤道子ファンクラブに入れてください。　　幸　俊威(17)大阪府

「安藤道子ファンクラブ」会長の荻窪圭氏による会則を次ページに示します。

1.会長は偉い
2.会長のみ抜け駆けを許される
3.安藤道子は上京するとき会長に連絡をする

ということですが，安藤さん，こんなこと野放しにしてていいんですか？

◆おお，安藤道子さんはすんげー美人ではないですか。最近，男ばかりの生活で（仕事の都合で）女性に対する免疫が弱くなっているのを痛感している私はXIGユーザーです。

棟方　正治(23)青森県

感動するのはいいですが，ちょっとこの文面は安藤さんに失礼だと思うぞ。

◆第19回データショウに行ってきました。シャープのブースではコンパニオンのねーちゃんが「これからのパソコンはシャープです！」といってました。やっぱりそうだったのか。前からそうだと思っていたけど。森　敦史(19)東京都

何をいまさら当たり前のことをいってるんでしょうかね。

◆第19回データショウへ行ってまいりました。いやあ，人，人，人の山で何が展示してあるのかさえよくわからないくらいでした。パソコン関係では，新しいノート型Mac（パワーブック）が人気を集めておりましたが，私はそれよりもMac ClassicIIに注目したい。68030（16MHz）2MバイトFDモデル（9インチディスプレイ一体型）で298,000円とは。X68000XVIももう少し安くなってもバチは当たらんと思うのですがねえ。

河野　浩(28)東京都

新しく発足した，AVCシステム事業推進室が鳥居さんを迎えたことで，どういった解答を用意しているのでしょうね。

◆この間，PC-9801用ハードディスクにと貯めていたお金が，X68000用の4Mバイト増設RAMに化けてしまった。そして，今度こそ買おうと思っていた矢先にビデオデッキが壊れた。僕のPC-9801はいつまでたっても標準のままだ。

相田　正彦(23)神奈川県

やっぱりかわいいのはX68000？

◆こんにちは。うちのCZ-8PP2に漢字ROMが入ってようやく漢字出力ができるようになりました。といっても，X68000ではサポートされていないのでテキスト出力するのも大変でした。というわけで，うれしくて編集部へのメッセージに使ってしまいました。ちゃんちゃん。こうやってみるとプロッタプリンタの文字も味わいがあると思いません？　私のようにペンプリンタを使ってメッセージを送ってきた人はいますか？　大島　靖浩(29)茨城県

でも，印刷されると活字になってしまってちょっと残念。

◆この前，近所のゲームセンターに行ったら「スターブレード」に4クレジット（400円）入っていた。しばらく様子を見て，誰も気がついていないようなのでなにくわぬ顔をして，STARTボタンを押してプレイしてしまった。ただでプレイしているので，かなり適当にやっていたつもりなのに，4クレジットでしっかりエンディングまでいってしまった。いつもなら5クレジットかかるのに，ただプレイの執念だろうか。

嶋田　裕文(20)大阪府

そいつはラッキーでしたね。

◆自宅に父親のMZ-5500があります（珍しいっていわれた）。今度パソコンをちゃんとやりたくて9月にX68000XVIを買いました。友人には「なんでX68000なんだ」といわれましたが，理由は単に「面白そうだから」です。まあ，これが高い買い物か安い買い物かこれからが勝負だと思っています。　南辺　玲(20)千葉県

親子揃ってシャープのパソコンに惚れ込むとはいい家族ですね。

◆Creative Computer Music入門(2)を読んで思ったのですが，コンピュータは音程がしっかりしているため和音を作る場合も理論を知っていると有利なんでしょうね。僕はアマチュアバンドでトロンボーンを吹いているんですけど，はっきりいってポジションなんてあるようでないし，気温とか体調で音程などはバリバリにかわっちゃうんです。そのぶん，合奏で自分が気持ちよく音が出せていると音楽が楽しくなってきます。音楽はこういった楽しさも重要だと思います。

安尾　文教(23)愛知県

デジタルであれアナログであれ，楽しいという気持ちは大切ですね。

◆今日，友達に風邪をうつされたみたいだ。人にうつせば早くよくなるだろうから，ハガキに向かってウイルスを飛ばしておこう。では，編集部の皆さん，ごきげんよう。

川田　宏(17)香川県

これを運んできた郵便局の人が風邪をひいたら君のせいだぞ。

◆私は香川県のある病院で透析技士をしているものです。店頭で知人友人から「PC-9801にしておいたほうがいいよ」という助言にもかかわらず，職場にX68000を導入しました。現在，CARD PRO-68Kを使用して患者のデータ解析をしています。購入の動機はいろいろありますが，やっぱり業務用アプリケーションが少ないのは悲しいですね。とはいえ仕事場のX68000XVIは最高です。一度取材してください。注射を1本サービスしてあげます。　富山　浩一(28)香川県

うっ，注射は嫌い。

◆僕は部活でアメフトをやっているんですが，この前試合中にケガ人が出てしまい，救急車がきたのです。入ってくるときにはグラウンドの真ん中を突っ切ってきたのに，出るときは遠慮したのかグラウンドの外側を通ろうとしたのです。そしたら，そしたら，なんとグラウンドの外側のところがぬかってて，救急車がその中にはまってしまったのです。皆は笑いをこらえながら救急車を救助して，見事抜け出した救急車は帰り際にクラクションを「プップッ」と鳴らしながらグラウンドから消えていきました。救急車のイメージを根底からくつがえす1日でした。　遊佐　勝(19)埼玉県

ちょっと珍しい光景ですね。

◆ハガキを出すのは3年ぶりぐらいです。最近はハガキを書くのが面倒だったから，長い間出していなかったけど11月号の「We Want You」を読んでから出さずにはいられなかった。同じような人は結構いるかもしれないので，今月はハガキがいっぱい返ってくると思いますよ。絶対に。　猶原　弘晃(21)福井県

うん。確かに11月号のハガキは10月号に比べてずいぶんと増えました。問いかけにきちんと答えてくれた人は本当にありがとう。ちゃんと目を通していますから。

◆11月号のmicroOdysseyを読んで思い出したことは，私が小学校3年生のときすでに「独身」に関する作文を書いていたということです。ガキの頃からスレていたんですね。ちなみに内容はというと「独身の利点と結婚の必要性」というとんでもないものでした。

小笠原　近広(19)東京都

小さい頃の作文って妙に恥ずかしいんですよね。

◆先日，ラジオを聞いていたら「VISTAのトランクは510リットルもあるのよ……」といったセリフが聞こえてきた。一瞬，なんじゃそりゃと思ったけど，要するに冷蔵庫と同じようなものだと気づいた。単位というのは量を伝えるのではなく，人を惑わせるために使われているらしい。私はもっと単位がほしい，単位違いか。

家田　貴之(22)愛媛県

◀尾澤　宏　兵庫県
よくわからないけど「アリガトウ・中嶋悟」なわけですね（なんだそりゃ）。個人的にはダイナブックを持って走り回る鈴木亜久里が好き。

◀東　寛　愛媛県
いいえ，力強いこのオッサン。X68000が留守にしている間，しっかりイラストの腕をみがいちゃいましょう。

インテル係数があるのも，ハードディスクのシークタイムが微妙なのも，皆単位のせいなのね。

◆古代米レポートその後。10月後半に入って縄文米はわずかながらに穂をつけ，緑色から茶色へ少しずつ熟しつつあります。穂の中身が空っぽのものもあって少し不安ですが。あと2，3週間くらいで収穫か？　なんとか成功させたいです。　迫田　賢一(40)大阪府

台風によるアクシデントにもめげず，元気に育ってよかったですね。来年もチャレンジするのかな。

◆半年前から足の親指にあるイボと戦っています。いわゆる「おでき」形のものでなく，ある1点だけ皮膚が硬くなり，よく見ると中に黒い（血の塊）が点々と見えるというものです。実は1年前からできていたのですが，受験やらなんやらで放っておいたものが中で根を伸ばしていたようです。そのため，治療（液体窒素で焼く）しても，すぐに別の場所にできてしまいます。もうすでに10回以上焼いています。いったいいつになったら収まるのだろう。あ〜あ。

荻野　友隆(18)京都府

おだいじに。

◆10月27日。「テレビ探偵団」を見ていたら「ラリホラリホラリルレロー」の出どころがわかってしまった。荻窪さんの年齢がしのばれる。23だな。　村木　俊之(18)神奈川県

ブッブー。はずれです。

◆最近家に住み着いているネズミがやけに元気でしょうがない。特に足の下でネズミが駆け抜けていくときの音は，サラウンドの比ではありません。一度きてもらえれば，この感触を実感してもらえるはずです。国辺　恭司(17)佐賀県

あの，小さな足がトトトトーッと駆け抜ける感触は，結構気持ちいいと思うけど。

◆ある日，家に帰ってみるとクロネコヤマトの不在票が入っていたので，さっそくクロネコヤマトに電話したときの話です。

僕：「あの〜不在票が入っていたんですけど」
ヤマト：「それではお名前と電話番号をどうぞ」
僕：「045-XXX-XXXXの佐川です」
ヤマト：「茶川さんですか？」
僕：「佐川です」
ヤマト：「茶川さん？」
僕：「(しばし考えて）佐川急便の佐川です」
ヤマト：「いたずら電話はよしてください！」
僕：「……」

佐川　浩一(20)神奈川県

似っ痛のぺ離間さんとか，負っ永久一倶さんとかも（苦しいなあ）誤解されそう。

◆Oh!X11月号で知ったこと。
1.アンディ・ウォーホルを有田さんが知っていたこと
2.E.O.さんに恋人がいること
3.あの本が高橋さんのものだったこと

鈴木　崇(18)東京都

12月号には何があったかな。

▲河野　純也　愛知県
勢いのある絵柄はいいけどなんかまとまりがないって感じ。構成をもっとよく考えるとだいぶよくなると思うぞ。

◆プレゼントNo.5の絵はもしかして篤見唯子様では？　関口　利彦(20)群馬県

大当たり！　ということで，いきなりカラーイラスト＆年賀状の募集をします。暑中見舞いでは，イラスト点数が少なくて実現できなかったため今度こそは絶対にやりたい。皆さんの力作をお待ちしてます。

# ぼくらの掲示板

●掲載ご希望の方は，官製ハガキに項目(売る・買う・氏名・年齢・連絡方法……)を明記してお申し込みください。
●ソフトの売買，交換については，いっさい掲載できません。
●取り引きについては当編集部では責任を負いかねます。
●応募者多数の場合，掲載できない場合もあります。
●紹介を希望されるサークルは必ず会誌の見本を送ってください。

## 仲間

★サークル“Stone-tone”では，X1,MZ-1500,PC-8801などを中心にした音楽活動をしています。活動内容は月1回の会誌の発行（ディスク会報も検討中）です。会誌の内容は音楽だけでなく，いろいろなことを載せていこうと考えています。あと，PC-E500（ポケコン）＋MIDIのコーナーも新設する予定なので，ユーザーの方は気軽に入会してください。というわけでコンピュータミュージックに興味のある方，作曲のできる方，暇な方はぜひ入会してください。〒223　神奈川県横浜市港北区箕輪町1-1-W101　日吉台学生ハイツ内W-560　西尾　将人(19)

## 売ります

★X1シリーズ用パーソナルテロッパ「CZ-8DT2」を5,000円以上で売ります。ほとんど使用していませんが箱がありません。説明書，付属品はすべてあります。官製ハガキに購入希望価格と連絡先を書いて下記の住所まで送ってください。〒381　長野県長野市中村274　轟　丈幸(21)

★MZ-2500用増設VRAMボード「MZ-1R27A」，辞書ROMボード「MZ-1R28A」を送料込み各8,000円で。MZ-6500A/B/C用80286ボード「MZ-1M11」(処理速度平均1.6倍アップ)を送料込み25,000円で売ります。それぞれ完動品，マニュアル，箱，付属品すべてありです。連絡は往復ハガキでお願いします。〒969-16　福島県伊達郡桑折町字西大隅76-1　半沢　勉(21)

★拡張I/Oボックス「CZ-8EB3」(拡張I/Oボード付き）を送料込み5,000円で売ります。まずは往復ハガキで連絡してください。〒981　宮城県仙台市青葉区新坂町11-18　小原　健一(18)

★24ドット熱転写プリンタ「CZ-8PC3」と黒リボン4本，カラーリボン5本をあわせて20,000円で売ります。ただし，一部ドット欠けが生じますがそのほかは完動です。もうひとつ，エプソン熱転写プリンタ用カットシードフィーダ「AS500CSF」を8,000円で売ります。こちらは完全に新品です。連絡は官製ハガキで。〒567　大阪府茨木市水尾1-13-13　陣山　達夫(21)

## 買います

★Oh!X 1988年3月号で発表されたX1用MIDIボードの完成品を10,000円で買います。連絡は官製ハガキで下記の住所までお願いします。〒223　神奈川県横浜市港北区箕輪町1-1-W101　日吉台学生ハイツ内W-560　西尾　将人(19)

★（株）音研のX68000用立体アニメーション作成ハード＋ソフト「立体くん（セット)」を探しています。詳細など当方にとって未知の部分が多いのでお持ちの方は，往復ハガキで連絡していただけないでしょうか。よろしくお願いします。〒514　三重県津市上浜町2-102　ABマンション3-C号　田村　晃一郎(17)

## バックナンバー

★Oh!Xの創刊号から1989年9月号までと1990年1，3月号を希望の価格で買います。多少の汚れは可。切り抜きは不可。官製ハガキで連絡してください。〒675-24兵庫県加西市和泉町759-3　増田　渡

# 編集室から from E・D・I・T・O・R

## DRIVE ON

このコーナーでは，本誌年間モニタの方々の意見を紹介しています。今月は11月号の内容に関するレポートです。

●3Dというのは不思議なもので，飛行物体が近づいてくると妙な現実味を覚えます。たとえワイヤーフレームでもスピード感や雰囲気にのまれ，恐ろしいほどリアルに迫ってくるのです。やはり3Dはグラフィックの美しさより雰囲気が大切なんだと思いました。

また，「立体空間の料理法」では，3Dゲームを作るうえでのヒントになりそうな記事で興味深く読みました。3Dというのはわかりにくく，私も具体的にどうすればいいのかわからないところがあります。しかし，大きな可能性を持っていることは確かでしょう。今後もプログラミングするうえでの注意点や基本的な3Dの扱い方を紹介してほしいですね。

宍戸 輝光(18) X68000 PRO 東京都

●新製品紹介の「F-Card GT」ですが，今回の記事を読んでこのソフトを買う気にはなれなかった。X68000にはあまり種類の多くないカード型データベースを「使ってみようかな」という気にさせるのはいいと思います。筆者の方もどの部分を説明するのか悩んだと思いますが，こういうソフトは何回かに分けて少しずつ説明していくのがいいでしょう。

また，11月号の表紙はとっても見づらい。目を引くかもしれないけど気をてらう必要はないと思う(子供が恐がっています)。内容で勝負なのだから表紙はもっと美しいほうがいいですね。最後に満開製作所の広告にあった電子ちゃんのお母さんが用意したねずみさんは，モデルさんがいるのかしら，娘が欲しがって困ってるんです。

野原 志貴乃(29) X68000 ACE 埼玉県

●「空間彷徨型ゲーム」といっても，要は自分が空間を感じられればそれでいい，と私は思っています。いくら多重スクロールしようとも，自機や弾が同一平面上にあるゲームは立体ゲームではないといえるし，逆に奥行きを感じさせてくれる「ウィザードリィ」は立派な立体ゲームでしょう。そして，空間というものは動きでも表現されるものだと思っています。フライトシミュレータだって，動きを止めてしまえばただの抽象画にすぎません。律儀に計算するのはかまいませんが，それらしく動いて見えるのが大切だと思います。

中村 健(21) X68000 ACE-HD,MSX2+ 埼玉県

●「第3回サイクロンCG大会」の記事を見て，どの作品も絵画として見た場合，光の使い方が下手ですね。全体的に明るすぎるという印象を受けます。また，機能を使ってみたという感じでテーマが感じられないのも問題でしょう。とはいってもなかなかきれいな作品が多いので，もっとほかの作品も見てみたい。いっそのことOh!X誌上でたくさんの作品を載せて，読者投票で賞を与える大会があれば面白いんじゃないですか。

また，11月号はS-OSの記事が2つもあってうれしいかぎりです。「Small-C活用講座」は関数コールの構造とかがすごく勉強になりました。「MORTAL」は，画面写真からの印象は強くありませんでしたがショッキングなストーリー，内容，使用したテクニックの解説まで載っているのがとてもよかったです。

佐藤 哲也(21) MZ-2500 北海道

●連載当初はどうなるのかよくわからなかった「ANOTHER CG WORLD」もかなり意図がはっきりしてきたようですね。浮かんだイメージを数字にしなければならないCGの世界において，作者のイメージと完成した作品をみる者とのイメージにギャップができやすいのではないかと思います。そのギャップを埋める意味で「ANOTHER CG WORLD」はかなり意味のあるページでしょう。

大手前高校理化学研究部 櫛田 宏樹(17) X68000 ACE,PC-9801RX,FM TOWNS,X1G/C,FM77AV,MSX2 大阪府

## ごめんなさいのコーナー

Oh！X Books「Z-MUSICシステム」

「Z-MUSICシステム」のマニュアルに不備な点がありました。以下に訂正事項を記載します。誠に申し訳ありませんでした。

P.4「XAPNEL.X」はディスク1，2，3いずれにも収録されていません。

P.23「@T」コマンドの説明

テンポとタイマ値の相関式でA/Bが入れ替わっています。

P.24「@N」コマンドの説明

文中の「10〜15」は「10〜25」の誤りです。

P.30「I」コマンドの説明

「I」コマンドでは，第2パラメータの省略ができません。

P.43「mtempo」の項の備考

「A0.L＝現在のタイマ値を書く」は「A0.L＝現在のタイマ値を返す」の誤りです。

P.48 ファンクション$43の説明が抜けていました。以下を追加します。

ファンクション$43 picturesync

機能　映像同期モードのオン/オフ

引数　D2.L＝0 モードオフ

　　　D2.L≠0 モードオン

備考　使用方法についてはP53参照

また，ディスクに収録されているZMUSIC.X，ZP.Xにバグがありました。リスト1，2を入力した後，実行すると画面に指示が出ます。それに従って，プログラム本体にパッチ当てを行ってください。念のため，その際にはバックアップを取ったシステムを使うようにしてください。

### リスト1

```
 10 /*          ZMUSIC.X書き換えプログラム
 20 int a
 30 print"準備が出来たら何かキーを押して下さい。"
 40 while (inkey$="")
 50 endwhile
 60 /*ドライブ名やファイル名は変更すること
 70 a=fopen("zmusic.x","rw")
 80 fseek(a,&H13FE,0):fputc(&H3D,a)
 90 fseek(a,&H1402,0):fputc(&H9E,a)
100 fseek(a,&H1403,0):fputc(&H1D,a)
110 fseek(a,&H8E20,0):fputc(&H6A,a)
120 fclose(a)
130 print "終了しました。"
140 end
```

### リスト2

```
 10 /*          ZP.X書き換えプログラム
 20 int a
 30 print"準備が出来たら何かキーを押して下さい。"
 40 while (inkey$="")
 50 endwhile
 60 /*ドライブ名やファイル名は各自変更すること
 70 a=fopen("zp.x","rw")
 80 fseek(a,&H54B,0):fputc(&H80,a)
 90 fclose(a)
100 end
```

**バグに関するお問い合わせは**

**☎03(5488)1311(直通)**

**月〜金曜日16：00〜18：00**

お問い合わせは原則として，本誌のバグ情報のみに限らせていただきます。入力法，操作法などはマニュアルをよくお読みください。

また，よくアドベンチャーゲームの解答を求めるお電話をいただきますが，本誌ではいっさいお答えできません。ご了承ください。

# 気になるZ-MUSIC発売その後

▶X68000の世界でも，ウィンドウ環境に対する期待は大きなものになっているようですが，それを受け止めるだけのパワーはまだ残念ながら持っていないようです。土台がまだ固まっていないからということもあるのでしょうが，それはそれとして，もっと周りから盛り立てるような動きがあってもいいはずです。ネットなどではいろいろとSX-WINDOW用のプログラムが出てきているようですが，やはり一般のユーザーからするとSX-WINDOWはたまに立ち上げるけど，“たいしたことはしていない”とか，“ウィンドウでしかできないことはやっていない”というのが率直な意見でしょう。アプリケーションが少ないというのが,いちばんの問題なのでしょうか？

▶前号とほぼ同時に(少し遅れましたが)，Oh!X Books「Z-MUSICシステム」を発売することができました。売れ行きもかなり順調なようで，スタッフ一同よろこんでいます。しかし，慣れないことでいろいろと不備もあり，お買い上げいただいた方々にご迷惑をかけたようです。内容に関しては，「ごめんなさいのコーナー」をご覧いただくとして，装丁のまずさからか，一部の商品においてディスクが壊れていたということもありました。壊れていたディスクはもちろん交換させていただいていますが，購入していただき中身が不良だった方の気持ちを考えると心苦しいばかりです。次に別冊を刊行するときにはこのようなことがないように十分に注意を払うつもりですので，どうかご了承ください。本当にもうしわけありませんでした。

▶ごめんなさいのコーナーには載っていないのですが，Z-MUSICのシステムディスクで立ち上げると，BASICのディレクトリにパスが通りません。これは単なるAUTOEXEC.BATでのタイプミスで，

PATH ……¥A:¥BASIC……

となっているところを，

PATH ……A:¥BASIC……

とすれば大丈夫です。

**投稿応募要領**

- 原稿には，住所・氏名・年齢・職業・連絡先電話番号・機種・使用言語・必要な周辺機器・マイコン歴を明記してください。
- プログラムを投稿される方は，詳しい内容の説明，利用法，できればフローチャート，変数表，メモリマップ（マシン語の場合）に，参考文献を明記し，プログラムをセーブしたテープ（ディスケット）を添えてお送りください。また，掲載にあたっては，編集上の都合により加筆修正させていただくことがありますのでご了承ください。
- ハードの製作などを投稿される方は，詳しい内容の説明のほかに回路図，部品表，できれば実体配線図も添えてください。編集室で検討のうえ，製作したハードが必要な場合はご連絡いたします。
- 投稿者のモラルとして，他誌との二重投稿，他機種用プログラムを単に移植したものは固くお断りいたします。

**あて先**

〒108 東京都港区高輪2-19-13 NS高輪ビル
ソフトバンク出版部
Oh!X「㋢㊀㋮㊔」係

# SHIFT・BREAK

▶駅の改札の横に黒板が出ていると，ついドキッとしてしまう。だいたいにおいて電車が止まっているとかの悪い話だからだ。で，編集部の最寄りの泉岳寺駅の改札にも「急告」という看板が出ていて，通るたびにドキッとしている。何が書いてあるかというと，「泉岳寺は反対の出口です」と意味不明なので，世の悩みの種は尽きないのである。（八）

▶我が家の近所に放置自転車の管理場所がある。主に駅前に放置された自転車たちが，役人の手によって連れ去られてきているのだ。乱雑に積み上げられ，野ざらしにされ，まるでそこは自転車たちのアウシュビッツだ。放置するほうが絶対的に悪いとはいえ，いったい何の権利があってこんなことをするのだ？まったく，見たくない光景である。（で）

▶文化の秋です。私もここらでいっちょ文化的な趣味を持とう，というわけでルーブル美術館展に行ってまいりました。いやあ，すごい混雑。子供連れからカップルからオヤジから，よくまあこんなに来るもんです。せめてドラクロワぐらいもうちょっとゆっくり観せてくれー。インテリ気取りになりそこなって，少し機嫌が悪くなった1日でした。（浦）

▶40度近い熱が3，4日続き，病院嫌いの私が薬をもらおうと，歩いて5分の救急病院に出かけた。ところが診断結果は肺炎。結局12日間，病院に泊まり込むはめになった。5時半に起床して，日中は点滴を4本も打たれ，夜は9時に就寝する生活はとても楽しかった。こんなことなら解熱剤を飲んでまで豊島園に行くんじゃなかった。（H.K.）

▶ZMUSICディスクⅠのSC-55用サンプル曲「SUMMER.BAS」はCメロのコードが合ってないし，収録されていないプログラムの説明があったり，バグがまだあったり，説明が抜けてたり，発売が遅れたり……，電話が鳴るたびビクビクしてる。でもいつもより時間をかけただけあって，昔X1用に作ったMusic BASICよりはいいものになったと思う。（善）

▶最近お気に入りのCMは「玄関開けたら2分でご飯」である（製品自体はおいしくないけど）。さて，宮沢りえの写真集を買ってしまった。きれいなだけで，色気はないし，動的でないし，迫力がない。田中康夫がいうには「自らは裸の王様でしかなかったという虚栄の決算」だそうで，私もそれに賛成である。ヌードになってもぶっとばなきゃ。（K）

▶大きめの地震があった翌日，ガスが出なくなった。ガス代は払っているのにと思いつつ，ガスメーターを調べたら何かのランプが点滅していた。そばの説明書に，ガスが止まったら近くのボタンでリセットしろと書いてある。要は安全装置が働いたのだった。これなら地震が来てもガスの元栓を閉める必要がないなと不謹慎なことを考えてしまった。（KO）

▶MI買った。現品処分で結構値引きしていたもんで，ついふらふらとお金を握りしめてレジへ直行してしまったのだ。冬のボーナスをあてにして，貯金の中からスカッと消えた十数万円。瀧氏に頼んでMIDIボードも買ったし（ケーブルだけはいまだない），さ〜てひまになったら思う存分遊びましょ。とうぶん先になるだろうけど。（J）

▶最近買って，よろこんでいるもの。リムーバブルハードディスクの本体とメディア2枚（AMIGAにSCSIでつないでいる）。「ポピュラスⅡ」の遊べるデモが付録についていた向こうの雑誌。買いかけて止めたもの。中古のAPPLEⅡc，39,800円ナリと価格不詳のAPPLEⅡGS。やっぱり，こんなものばっかり買ってよろんこんでいちゃあいかんな。（A）

▶どうも最近頭が休まる暇がないな。通勤中ぐらいボーッとしていたいのに，突然いいフレーズが浮かんだりするもんだから，結局1曲作っちゃうはめになるし。家に帰ったら帰ったで寝りゃいーものを，衝撃にかられてダーッと小説書いちゃったりするし。おまけに，いままで夢を記憶していたことなんて皆無なのに，最近はしっかり。ああ，もう！（E.O.）

▶今月はすべて雷語でやってみようとしたがだめだった。せめて行数表示は必要だよなあ。WP.Xがないと本が出ないのはウチくらいのもんだろうか。AI辞書が賢いので少し救われてはいるが……。さて，年末といえば第九。第九は速い曲だが，CDではみんなのたのたしていて気に入らない。ミュンシュより速い第九ってのはないのだろうか？（U）

▶ご近所のMacintoshをのぞき込むと，そこにはあの「ぎょぴちゃん」が泳いでいる。「ぼてち」をあげるとニコッと笑うのだ。リソースを書き換えて，ウィンドウがきんぎょばちの形になっているらしい。SX-WINDOWでもできないかな。ちなみに「千歳」というのは私の通っていた高校の名前です。だからなんだといわれても困るけど。（T）

## microOdyssey

4年前の夏，イラストを描き始めた。おりしもそのときは現役受験生，しかも夏休みとくればまだまだ緊迫とは無縁の状態（？）だったので，机に向かっている時間が多いわりにそれほど勉強がはかどっていなかった。要するに時間を持て余すことがよくあったのだ。

最初のうちはノートの余白にラクガキ程度のものを，ちびちび描いては自己満足に浸っていた。しばらくすると，やっぱり他人の評価がほしくなる。机の中を引っ掻き回して探し出したペン（たぶんカブラペンだった）と墨汁を手に取り，禁断の投稿への道を歩み始めた。

なにしろ，始めたばかりでわからないことだらけ。資料用と称して昼食代を使い込みマンガとそのテの雑誌を買いあさり，本に載っていた話を読むたびに画材屋へ通う毎日。ここまでくると「もっとうまくなりたい」という欲が出てくるようになり，道具を揃え試行錯誤でイラストを描き続ける。使用インクやペンを変え，紙もハガキからケント紙などいろいろ使ってみた。そうして，いままでわからなかったことが理解できるようになっていく。やがてそんな過程が面白いと感じるようになり始めていった。

いま思えばこの頃がいちばん楽しんでイラストを描いていたように思う。好きなときに好きなことを考え，頭の中にあるイメージを膨らましていきペンと紙を使って表現する。力の入れ具合ひとつで成功したり失敗したり，喜んだり怒ったり，気づいたことを片っ端から実践し，時間をかけて取り組んできた。

また，うまく描き上げようとずいぶんあがいたこともある。途中で投げ出しそうになったこともあるが，適度なこだわりを持ち，ひとつの趣味として力まず自分の思いのままを描いていこうという気持ちはずっと抱き続けている。自由な発想で楽しむことができないのなら，それは趣味といえないだろう。

ところが，この自由な発想というものが結構くせものである，というのは読者の皆さんも十分承知しているだろう。なにかを作り出すためにはなにかしら手段，道具が必要となるからだ。つまり，自分の思考を表現する場合にそれは道具を通して行われ，使用する道具の制限を受けるということ。基本的にはどんなものでも表現は可能といえるが，そのための道具が見つからなかったり手段があやふやなときがある。ましてや，いままでやったこともない，まったく新しい分野のときにはかなり戸惑うことだろう。

1つひとつ問題を潰しながら目標に向かっていくことも大切だが，煮詰ってしまったら少し寄り道をするぐらいの余裕もほしい。時間は有限であっても，最終的に自分の満足できるものを作りたいなら妥協はしたくない。趣味でやっているのだから妥協は少なくしたいし，妥協のおかげでせっかくの発想が台なしになってしまう可能性もある。ここはやっぱり，アマチュアとしてのこだわりを持っていきたいものだ。

そうして，目標を達成したあとには十分な満足感と次なる目標が現れる。また，まったく別の道に踏み込むことも楽しい。

自由気ままにやりたい放題，この言葉が僕は大好きだ。「プロを目指すよりもセミプロのままのほうがいい」そんなアマチュア的な思考が僕の中にあるのかもしれない。（J）

## 1992年2月号1月18日(土)発売

**特集 2Dグラフィックの拡張**

・拡張版Z's-EX/画像処理プログラム

Oh! X GAME OF THE YEAR'91ノミネート発表

X68000ソフトウェアの傾向と対策

CARDDRV用カードゲーム Lucky

詳報 PressConductorPRO-68K/MIRAGE MODEL STAFF

## バックナンバー常備店

| | | |
|---|---|---|
| 東京 | 神保町 | 三省堂神田本店5F 03(3233)3312 |
| | 〃 | 書泉ブックマートB1 03(3294)0011 |
| | 〃 | 書泉グランデ5F 03(3295)0011 |
| | 秋葉原 | T-ZONE 7Fブックゾーン 03(3257)2660 |
| | 八重洲 | 八重洲ブックセンター3F 03(3281)1811 |
| | 新宿 | 紀伊国屋書店本店 03(3354)0131 |
| | 高田馬場 | 未来堂書店 03(3200)9185 |
| | 渋谷 | 大盛堂書店 03(3463)0511 |
| | 池袋 | リブロ池袋店 03(3981)0111 |
| | 〃 | 西武百貨店9F コンピュータ・フォーラム 03(3981)0111 |
| 神奈川 | 横浜 | 有隣堂横浜駅西口店 045(311)6265 |
| | 〃 | 有隣堂ルミネ店 045(453)0811 |
| | 藤沢 | 有隣堂藤沢店 0466(26)1411 |
| 神奈川 | 厚木 | 有隣堂厚木店 0462(23)4111 |
| | 平塚 | 文教堂四の宮店 0463(54)2880 |
| 千葉 | 柏 | 新星堂カルチェ5 0471(64)8551 |
| | 船橋 | リブロ船橋店 0474(25)0111 |
| | 〃 | 芳林堂書店津田沼店 0474(78)3737 |
| | 千葉 | 多田屋千葉セントラルプラザ店 0472(24)1333 |
| 埼玉 | 川越 | 黒田書店 0492(25)3138 |
| | 川口 | 岩渕書店 0482(52)2190 |
| 茨城 | 水戸 | 川又書店駅前店 0292(31)0102 |
| 大阪 | 北区 | 旭屋書店本店 06(313)1191 |
| | 都島区 | 駸々堂京橋店 06(353)2413 |
| 京都 | 中京区 | オーム社書店 075(221)0280 |
| 愛知 | 名古屋 | 三省堂名古屋店 052(562)0077 |
| | 〃 | パソコンΣ上前津店 052(251)8334 |
| | 刈谷 | 三洋堂書店刈谷店 0566(24)1134 |
| 長野 | 飯田 | 平安堂飯田店 0265(24)4545 |
| 北海道 | 室蘭 | 室蘭工業大学生協 0143(44)6060 |

## 定期購読のお知らせ

Oh!Xの定期購読をご希望の方は綴じ込みの振替用紙の「申込書」欄にある「新規」「継続」のいずれかに○をつけ，必要事項を明記のうえ，郵便局で購読料をお振り込みください。その際渡される半券は領収書になっていますので，大切に保管してください。なお，すでに定期購読をご利用の方には期限終了の少し前にご通知いたします。継続希望の方は，上記と同じ要領でお申し込みください。

**海外送付ご希望の方へ**

本誌の海外発送代理店，日本IPS（株）にお申し込みください。なお，購読料金は郵送方法，地域によって異なりますので，下記宛必ずお問い合わせください。

日本IPS株式会社
〒101 東京都千代田区飯田橋3-11-6
☎03(3238)0700


Oh!X 1月号
■1992年1月1日発行 定価600円(本体583円)
■発行人 孫 正義
■編集人 橋本五郎
■発売元 ソフトバンク株式会社
■出版事業部 〒108 東京都港区高輪2-19-13 NS高輪ビル
Oh!X編集部 ☎03(5488)1309
出版営業部 ☎03(5488)1360 FAX 03(5488)1364
広告営業部 ☎03(5488)1365
■印 刷 凸版印刷株式会社

# 満開の電子(でんこ)ちゃん

案：満開製作所
え：岡村祭

あけまして
おめでとう
ございます
満開の
電子です

いつも
同じセリフ
ばかりで
ごめんなさい
今回はお正月
らしく すごろく
形式に
してみました

でも
言ってる事は
いつもと同じだね

ここから
スタート
してね

**スタート**

ちょっと弱気になっていた
雨の夜 郵便振替の用紙
にうっかり住
所・氏名を記
入してしまう

システム入りで
すぐ起動
3コマ進む

キーボードに
コーヒーこぼして
18コマ戻る

停電で
データが
ふっとんだので
踊る

ウィルス混入
65,536回
休み

身長168cm
安田成美

ここに止まった人は
「電子ちゃんの単行本は
まだですか係」(満開製作
所・内）に
ハガキを送ること

もはや
月1度 水色の封筒が届かないと
いられないカラダになってしまった
人のみ 5コマ進む

バグが出た
1回休み

**ゴール**

色々あったが
満開製作所に入社
一生を棒に振る

いろいろ
あった…

グラフィックやツールの
充実に驚くわ
BEEP音で笑うわ
おまけに操作は
マウスで楽々
はらたいらさんに
3000点

「変酋長の小屋」を
読んで 祝一平
さんのファンになり
5コマ
進む

あげますコーナーで
当選 トド肉の
缶詰もらって
6コマ進む

ディスク
クラッシュで
ふり出しに戻る

間違って
ファイル削除
3コマ戻る

ハイッ社長
宮尾すすむ

うひゃ〜

購読方法：定期購読もしくはソフトベンダー武尊(タケル)でお買い求めいただけます。
★定期購読の場合＝定期購読料6ヶ月分6,000円（送料サービス、消費税込）を、
現金書留または郵便振替で下記の宛先へお送り下さい。
現金書留の場合：〒171 東京都豊島区要町1―19―3 いさみビル4F 満開製作所
郵便振替の場合：東京5―362847 満開製作所
●御注文の際は、郵便番号・住所・氏名・電話番号を忘れずに記入して下さい。
●新たに購読を開始される方は、「新規」とご明記下さい。
●製品の性格上返品には応じられませんが、お申し出があれば定期購読を解約し残金をお返しします。
★武尊でお求めの場合＝1部につき1,200円（消費税込）です。
●定期購読版と内容が一部異なる場合があります。ご了承下さい。
●お問い合わせ先 TEL(03)3554―9282（月～金 午前11時～午後6時）
（なお、定期購読版のバックナンバーについては定期購読者の方のみご注文を承ります）

進藤慎一
（青森県）

電脳倶楽部はひどい雑誌です。役に立つツールを載せて、私のシステムディスクを巨大にさせた挙げ句にハードディスクまで買わせたばかりか、面白いゲームを載せて私の部屋を友人たちの溜り場にしたり、笑えるビープ音で夜中に意味もなく笑わせてくれたり、音楽、グラフィックの保存のために貴重な磁性面を使わせたりと、本当にこんなにひどい雑誌は世界中探してもないでしょう。
もし私の話が信じられないならあなたが実際に定期購読してみて私の話が嘘でないことを確かめてください。

R&R 年末ゲームフェアー

X68000

# 今、下記セットをお買い上げのお客様には

| X68000 XVI | | |
|---|---|---|
| CZ-634C-TN | 定価 | ¥368,000 |
| CZ-606D-TN | 定価 | ¥ 79,800 |
| 合　計 | | ¥447,800 |
| R&R特価 | | ¥355,000 |

| X68000 SUPER | | |
|---|---|---|
| CZ-623C-TN | 定価 | ¥498,000 |
| CZ-606D-TN | 定価 | ¥ 79,800 |
| 合　計 | | ¥577,800 |
| R&R特価 | | ¥335,000 |

| X68000 PROⅡ | | |
|---|---|---|
| CZ-604C-TN | 定価 | ¥348,000 |
| CZ-606D-TN | 定価 | ¥ 79,800 |
| 合　計 | | ¥427,800 |
| R&R特価 | | ¥232,000 |

# お好きなゲームソフト3本をプレゼント!!

もしくはMS-DOS Ver3.3C基本セットとお好きなゲームを2本プレゼント致します。

**コンピュータグラフィックセット**

| 品名 | | 価格 |
|---|---|---|
| CZ-634C-TN | 定価 | ¥368,000 |
| CZ-606D-TN | 定価 | ¥ 79,800 |
| CZ-8PC5 | 定価 | ¥ 96,800 |
| Z'S STAFF PRO-68K V2.0 | 定価 | ¥, 58,000 |
| 定　価 | | ¥602,600 |
| R&R特価 | | ¥448,000 |

**パソコン通信セット**

| 品名 | | 価格 |
|---|---|---|
| Z-634C-TN | 定価 | ¥368,000 |
| CZ-606D-TN | 定価 | ¥ 79,800 |
| MD24FB5V | 定価 | ¥ 39,800 |
| たーみのる2 | 定価 | ¥ 17,800 |
| 合　計 | | ¥505,400 |
| R&R特価 | | ¥378,000 |

**コンピュータミュージックセット**

| 品名 | | 価格 |
|---|---|---|
| CZ-634C-TN | 定価 | ¥368,000 |
| CZ-606D-TN | 定価 | ¥ 79,800 |
| SX-68M | 定価 | ¥ 19,800 |
| CM-64 | 定価 | ¥129,000 |
| MA-12C(2台) | 定価 | ¥ 28,000 |
| Mu-1 SUPER | 定価 | ¥ 39,800 |
| 合　計 | | ¥664,400 |
| R&R特価 | | ¥498,000 |

**コンピュータミュージック**

| 品名 | | 価格 |
|---|---|---|
| SC-55 | 定価 | ¥ 69,000 |
| SX-68M | 定価 | ¥ 19,800 |
| Mu-1 SUPER | 定価 | ¥ 39,800 |
| 合　計 | | ¥128,600 |
| R&R特価 | | ¥102,000 |

**周辺機器**

| 品名 | 定価 | R&R特価 |
|---|---|---|
| ●プリンタ関係 | | |
| IO-735XB | ¥248,000 | ¥169,000 |
| IO-73SF(カットシートフィーダ) | ¥ 80,000 | ¥ 63,000 |
| JX-220/B(イメージスキャナー) | ¥145,000 | ¥113,000 |
| ●増設メモリー | | |
| CZ-6BE1 | ¥ 35,000 | ¥ 27,500 |
| PIO-6EB1A | ¥ 25,000 | ¥ 19,500 |
| PIO-6BE4 | ¥ 88,000 | ¥ 69,000 |
| ●その他のオプション | | |
| CZ-6BS1 | ¥ 29,800 | ¥ 23,800 |
| CZ-8NJ2(サイバスティック) | ¥ 23,800 | ¥ 19,000 |

※その他周辺機器も取り扱っていますので、お気軽にお問い合わせ下さい。

## ▼ソフト通信販売コーナー ソフトが最大25%OFF・消費税サービス▼

(ただし、エニックス・シャープ・電波新聞・ポニーキャニオン等一部のメーカーは除きます。)

| タイトル | 定価 |
|---|---|
| F15ストライクイーグル | ¥10,800 |
| Z'S STAFF PRO68K | ¥58,000 |
| アークスオデッセイ | ¥ 8,800 |
| アクアレス | ¥ 8,700 |
| アルシャーク | ¥ 9,800 |
| イース | ¥ 9,600 |
| イメージファイト | ¥ 9,700 |
| インペリアルフォース | ¥ 8,800 |
| エメラルドドラゴン | ¥ 9,800 |
| キャメルトライ | ¥ 8,800 |
| サーク | ¥ 8,800 |
| 飛翔鮫 | ¥ 8,800 |
| サイバーコア | ¥ 7,800 |
| サイレントメビウス | ¥14,800 |
| シムアース | ¥12,800 |
| シムシティ | ¥ 8,800 |
| ジェノサイドⅡ | ¥ 8,800 |
| スーパー大戦略 | ¥ 8,800 |
| ゼノンⅡ | ¥ 9,700 |
| デッドアッシュ | ¥ 8,800 |
| ドラッケン | ¥ 8,800 |
| ネクタリス | ¥ 8,800 |
| パロディウスだ! | ¥ 9,800 |
| パワーモンガー | ¥12,800 |
| ファーストクイーンⅡ | ¥ 8,800 |
| ファランクス | ¥ 8,800 |
| フェアリーランドストーリー | ¥ 6,800 |
| プリッツ・クリーク | ¥ 9,800 |
| プルトンレイ | ¥ 8,800 |
| プリンスオブペルシャ | ¥ 9,800 |
| プロサッカー68 | ¥ 9,800 |
| ボナンザブラザース | ¥ 9,000 |
| ボンバーマン | ¥ 7,800 |
| ポピュラス | ¥ 9,800 |
| メルヘンメイズ | ¥ 8,800 |
| ラグーン | ¥ 8,800 |
| ラプラスの魔 | ¥ 8,700 |
| ロードス島戦記 | ¥ 9,800 |
| スターウォーズ | ¥ 7,200 |
| 銀河英雄伝説Ⅱ DX SET | ¥12,800 |
| 出たな!!ツインビー | ¥ 9,800 |
| 大戦略Ⅲ'90 | ¥ 9,800 |
| 遥かなるオーガスタ | ¥12,800 |

◉その他ソフトも取り扱っています。
◉お気軽にお問い合わせください。

※掲載商品の価格は、全て消費税別です。

## 通信販売でのお申し込み方法

**ハードの場合**

★商品は電話またはファックス(お客様の電話番号をお忘れなく)でご注文下さい。
★お支払いは銀行振込でお願いします。入金確認後の発送となります。
　ソフトに関しては現金書留も可能です。ローンも取り扱っています。
★特に記載がない商品には送料・消費税は含まれておりません。
★掲載以外にも各社商品を扱っておりますので、お気軽にご相談下さい。
　振込先:富士銀行 西大井支店(普)1358191 アール・アンド・アール・メディア㈱

**ソフトの場合**

お客様がお求めになる商品の定価の合計によって下記の通りサービス(割引率)が変わります。
なお、メンバーズカードは今商品お買い上げのお客様に無料プレゼント中です。(通信販売も含みます。)

サービス内容 (消費税はサービス。( )内はメンバーズカードをお持ちの場合)

| 定価合計が | (メンバーズカード) | 割引 | 送料 |
|---|---|---|---|
| 5千円未満 | | 15%OFF | 送料300円 |
| 5千円以上1万円未満 | (5千円未満) | 20%OFF | 送料500円 |
| 1万円以上1万5千円未満 | (1万円未満) | 20%OFF | 送料サービス |
| 1万5千円以上2万円未満 | (1万5千円未満) | 23%OFF | 送料サービス |
| 2万円以上 | (1万5千円以上) | 25%OFF | 送料サービス |

●例:商品の定価合計が12,800円の場合10,240円(20%OFF、送料・消費税サービス)になります。
●メンバーズカードをお持ちですと9,856円(23%OFF、送料・消費税サービス)になります。

☞今、R&Rメディアで15,000円以上商品を買うと特製テレホンカードをプレゼント致します。

パソコンショップ

# R&Rメディア

☎03-3777-7335
fax.03-3777-6448
〒140 東京都品川区西大井6-10-10 品川RSビル
営業時間/11:00～20:00(火曜日定休日)

各種教室及びソフト体験コーナー開講中!!
詳しくはR&Rまでお気軽にお尋ね下さい。

◉取扱い商品 NEC・富士通・エプソン・シャープ
(メーカー保証付) ソフト、各種サプライ用品

株式会社 デンキヤ

〒332 埼玉県川口市西川口4丁目6番4号
AM11:00〜PM7:00 無休

## 今月の超特価品

シャープ
X68000セット
XVI

特価 299,700円より各種

TEL 0482-54-3400

### ★X6800本体★

| 品番 | 価格 |
|---|---|
| CZ-644C-TN | ¥ |
| CZ-634C-TN | ¥ |
| CZ-653C | ¥ 192,400 |
| CZ-623C-TN | ¥ 323,700 |
| CZ-604C-TN | ¥ 226,200 |

### ★X6800ディスプレイ★

| 品番 | 価格 |
|---|---|
| CZ-607D | ¥ 68,400 |
| CZ-614D | ¥ 91,100 |
| CZ-606D | ¥ 53,100 |
| CZ-604D | ¥ 64,000 |
| CU-21HD | ¥ 99,900 |

### ★プリンタ・ケーブル付★

| 品番 | 価格 |
|---|---|
| CZ-8PG1 | ¥ 90,400 |
| CZ-8PG2 | ¥ 111,200 |
| CZ-8PK10 | ¥ |
| CZ-8PC5 | ¥ 67,300 |
| IO-735X | ¥ |
| CZ-6PV1 | ¥ |

### ★RAMボード★

| 品番 | 価格 |
|---|---|
| CZ-6BE1B | ¥ 21,000 |
| CZ-6BE2 | ¥ |
| CZ-6BE4 | ¥ |
| PIO-6BE1-A | ¥ 18,100 |
| PIO-6BE2 | ¥ 33,800 |
| PIO-6BE4 | ¥ 59,400 |
| CZ-6BE2A | ¥ 44,900 |
| CZ-6BE2B | ¥ 41,000 |

### ★その他★

| 品番 | 価格 |
|---|---|
| CZ-6BP1 | ¥ |
| CZ-6EB1 | ¥ |

### ★ハードディスク各種★

| 品番 | 価格 |
|---|---|
| CZ-64H | ¥ 90,000 |
| TX-80 | ¥ 79,000 |
| TX-130 | ¥ 99,800 |

### ★インターフェイス各種★

| 品番 | 価格 |
|---|---|
| CZ-6BS1 | ¥ 22,400 |
| CZ-6BM1 | ¥ 20,100 |
| CZ-6BV1 | ¥ 15,800 |
| CZ-6BF1 | ¥ |
| CZ-6BG1 | ¥ |
| CZ-6BU1 | ¥ |
| CZ-6BC1 | ¥ |
| CZ-6BL1 | ¥ |
| CZ-6BL2 | ¥ |
| CZ-6BP2 | ¥ |

### ★周辺機器各種★

| 品番 | 価格 |
|---|---|
| CZ-8NJ2 | ¥ 17,900 |
| CZ-8NJ1 | ¥ 1,300 |
| CZ-8NM3 | ¥ 7,400 |
| CZ-8NT1 | ¥ 10,400 |
| CZ-8NM2A | ¥ 5,100 |
| BF-68PRO | ¥ 13,800 |
| CZ-6TU-BK | ¥ 23,000 |
| CZ-6VT1 | ¥ 48,500 |
| CZ-6SD1 | ¥ |

### ★モデム各種★

| 品番 | 価格 |
|---|---|
| MD24FB5V | ¥ 28,900 |
| PV-M24B5 | ¥ 27,700 |
| PV-A24B5 | ¥ 27,700 |
| コムスターズ2424/5 | ¥ 25,500 |
| コムスターズ2424/4 | ¥ 24,000 |

### ★ソフト各種★

| 品番 | 価格 |
|---|---|
| CZ-249GS | ¥ 22,400 |
| CZ-255GS | ¥ 6,600 |
| CZ-256GS | ¥ 6,600 |
| CZ-245LS | ¥ 33,600 |
| CZ-260LS | ¥ 7,400 |
| CZ-251BS | ¥ 29,900 |
| CZ-243BS | ¥ 14,900 |
| CZ-240BS | ¥ 11,100 |
| CZ-278SS | ¥ 7,400 |
| CZ-257CS | ¥ 14,900 |
| CZ-219SS | ¥ 22,400 |
| CZ-252MS | ¥ 21.600 |
| CZ-213MS | ¥ 14,100 |
| CZ-247MS | ¥ 21.600 |

### ★ゲームソフト各種★

| 品名 | 価格 |
|---|---|
| シグナトリー | ¥ 8,900 |
| パロディウスだ | ¥ 7,350 |
| FOXY2 | ¥ 5,800 |
| まぁじゃん2 | ¥ 5,800 |
| 遥かなるオーガスタ | ¥ 9,400 |
| ファランクス | ¥ 5,800 |
| 生中継68 | ¥ 7,400 |
| サイレント メビウス | ¥ 11,500 |
| A列車で行こうIII | ¥ 11,500 |
| シムシティー | ¥ 7,350 |
| スコルピウス | ¥ 5,800 |

24時間テレホンサービス
0482-54-3444

## お申し込みはお電話で

TEL 0482-54-3400
FAX 0482-54-3443

★振込先★
三菱銀行西川口支店
普通 0258081
㈱デンキヤ

西川口駅
至南浦和
西口より
徒歩8分
至川口

# XVI X68000シリーズ
# 年末謝恩セール中!!

## 全国送料 無料サービス実施中（1万円以上お買い上げの方）

| 型番 | 品名 | 定価(円) | 特価(円) |
|---|---|---|---|
| AC-02 | PC-9801用キーボード延長ケーブル(1.5m) | 2,500 | 2,000 |
| AC-06 | アナログディスプレイ延長ケーブル15P(1.5m) | 4,800 | 3,900 |
| AC-10 | X68000用キーボード延長ケーブル(1.5m) | 2,500 | 2,000 |
| AN-1506 | 15ピン6ピンディスプレイ変換ケーブル | | 1,700 |
| AN-1508 | 15ピン8ピンディスプレイ変換ケーブル | | 1,700 |
| AN-S100 | アンプ付スピーカー | 36,600 | 特価 |
| AN-X68 | キーボードシリコンカバー(EXPERT) | 3,500 | 2,800 |
| BF-68PRO | ディスプレイフィルター | 19,800 | 16,800 |
| CE-120P | ポケコンプリンタ | 24,800 | 21,800 |
| CE-123P | ポケコンプリンタ | 19,800 | 17,800 |
| CE-124 | カセットインターフェイス | 4,500 | 3,600 |
| CE-126P | ポケコンプリンタ | 17,800 | 13,800 |
| CE-130T | RS-232Cコンバータ | 17,800 | 15,800 |
| CE-135T | PC-E500/550用RS-422Cコンバータ(Mac用) | 9,800 | 8,800 |
| CE-140F | ポケットディスクドライブ | 49,000 | 44,800 |
| CE-140T | PC-E500/550用RS-232Cコンバータ(PC用) | 9,800 | 8,800 |
| CE-159 | 8KBプログラムモジュール | 35,000 | 4,200 |
| CE-1600E | パラレルポケットディスクI/F | 19,800 | 17,800 |
| CE-1600F | ポケットディスクドライブ | 39,800 | 34,800 |
| CE-1600M | 32KBプログラムモジュール | 32,000 | 16,000 |
| CE-1600P | 4色カラープロッタプリンタ | 69,800 | 59,800 |
| CE-1601M | 64KBプログラムモジュール | 45,000 | 30,000 |
| CE-161 | 16KBプログラムモジュール | 50,000 | 3,800 |
| CE-1650F | ポケットディスク(10枚組) | 9,800 | 8,800 |
| CE-201M | 8KB増設RAM | 18,000 | 3,000 |
| CE-202M | 16KB増設RAM | 35,000 | 6,000 |
| CE-203M | 32KB増設RAM | 32,000 | 7,000 |
| CE-212M | 8KB増設RAM | 18,000 | 6,000 |
| CE-2H16M | 16KB増設RAM | 16,000 | 11,000 |
| CE-2H32M | 32KB増設RAM | 32,000 | 16,000 |
| CE-2H64M | 64KB増設RAM | 45,000 | 30,000 |
| CE-T800 | PC-E200用RS-232Cコンバータ | 12,800 | 11,800 |
| CU-14FD | アナログカラーディスプレイ0.31ドットピッチ | 74,800 | 特価 |
| CU-14KD | アナログカラーディスプレイ0.28ドットピッチ | 89,800 | 特価 |
| CU-14TV | デジタル/アナログディスプレイTV(0.31) | 98,800 | 64,800 |
| CU-SX1 | パソコン用プロジェクターキケーブル15P(3m) | 980,000 | 800,000 |
| CW-100 | ICカードリーダー | 40,000 | 36,000 |
| CW-100D | 開発マニアル | 40,000 | 36,000 |
| CZ-115LF | X1FORTRAM(言語) | 13,800 | 11,700 |
| CZ-116LF | X1C(言語) | 13,800 | 11,700 |
| CZ-128SF | X1CP/M(OS) | 9,800 | 8,500 |
| CZ-130SF | X1turbo漢字CP/M(OS) | 14,800 | 12,500 |
| CZ-141F | NEW ZBASIC | 18,800 | 16,000 |
| CZ-141SF | X1turbo NEW BASIC | 18,800 | 16,000 |
| CZ-300F | 3"フロッピーディスクドライブ | 79,800 | 6,000 |
| CZ-31F | 300F用増設ドライブ | 59,800 | 6,000 |
| CZ-3CP/M | 3インチ X1 CPM | | 5,000 |
| CZ-501H | X1増設用ハードディスクユニット | 258,000 | 60,000 |
| CZ-612DGY | 15"カラーディスプレイTV | 119,800 | 80,000 |
| CZ-6BC1 | X68000FAXボード | 79,800 | 67,800 |
| CZ-6BE1B | 1MB増設RAMボード | 28,000 | 19,500 |
| CZ-6BE2 | 2MB増設RAMボード | 79,800 | 63,800 |
| CZ-6BE4 | 4MB増設RAMボード | 138,000 | 110,400 |
| CZ-6BF1 | RS-232C増設ボード | 49,800 | 42,300 |
| CZ-6BG1 | X68000GPIBボード | 59,800 | 50,000 |
| CZ-6BM1 | MIDIボード | 26,800 | 23,800 |
| CZ-6BN1 | スキャナボード | 29,800 | 25,300 |
| CZ-6BP1 | 数値演算ボード | 79,800 | 63,800 |
| CZ-6BS1 | SCSIボード | 29,800 | 23,800 |
| CZ-6BU1 | ユニバーサルI/Oボード | 39,800 | 33,800 |
| CZ-6BV1 | ビデオボード | 21,000 | 16,000 |
| CZ-6PV1 | ビデオプリンタ | 198,000 | 158,000 |
| CZ-6SD1 | システムラック | 44,800 | 36,000 |
| CZ-6TU | RGBシステムチューナー | 33,100 | 26,500 |
| CZ-820C | X1G MODEL10 | 69,800 | 16,800 |
| CZ-822C | X1G MODEL30 | 118,000 | 39,800 |
| CZ-82F | CZ-802C用増設ドライブ | 59,800 | 6,000 |
| CZ-830C | X1 twin | 99,800 | 35,000 |
| CZ-8BF1 | フロッピーディスクI/F | 14,800 | 11,500 |
| CZ-8BCR2 | グラフィックボードX1 | 14,800 | 3,000 |
| CZ-8BK2 | X1漢字ROM | 19,800 | 16,800 |
| CZ-8BM2 | RS-232Cマウスボード | 19,800 | 16,800 |
| CZ-8BO1 | フロッピーディスクI/F | 14,800 | 8,000 |
| CZ-8BS1 | X1FM音源ボード | 23,800 | 19,500 |
| CZ-8BV2 | カラーイメージボード | 39,800 | 32,000 |
| CZ-8EB3 | X1拡張I/Oボックス | 33,800 | 28,000 |
| CZ-8LM1 | RS-232Cケーブル | 7,200 | 6,000 |
| CZ-8LM2 | RS-232Cクロスケーブル | 7,200 | 6,000 |
| CZ-8NJ1 | ジョイカード | 1,700 | 1,360 |
| CZ-8NJ2 | インテリジェントコントローラ | 23,800 | 18,500 |
| CZ-8NS1 | カラーイメージスキャナ | 188,000 | 149,000 |
| CZ-8NT1 | トラックボール | 13,800 | 11,500 |
| CZ-8PC5 | 48ドット熱転写漢字プリンタ | 96,800 | 特価 |
| CZ-8PK10 | 24ドット136桁漢字プリンタ | 99,800 | 特価 |
| CZ-8PK7 | 24ドット80桁漢字プリンタ | 122,000 | 59,800 |
| CZ-8TM1 | モデムユニット300bps(X1ソフト付属) | 29,800 | 5,000 |
| CZ-8TM2 | モデムユニット300/1200bps(X1ソフト付属) | 49,800 | 39,800 |
| CZ-8WB51 | X1ターボ disk BASIC | 9,800 | 4,000 |
| HXD040 | ITEM 40MBHD(SACI外付型) | 118,000 | 65,000 |
| HXD140 | ITEM 40MBHD(SACI内蔵型) | 98,000 | 69,000 |
| IO-735X | カラーイメージジェットプリンタ | 248,000 | 180,000 |
| IP-1251 | MZ-2800デスクUP | 88,000 | 10,000 |
| IP-1253 | MZ-2800クリッパー | 77,000 | 10,000 |
| IP-1254 | MZ-2800プランUP | 66,000 | 10,000 |
| JX-100 | ハンディカラースキャナ | 89,800 | 特価 |
| LED看板 | 15cm＊15cm＊8文字3色 | 950,000 | 特価 |
| MZ-1D10 | 12"モノクロディスプレイ | 41,800 | 25,000 |
| MZ-1D17 | 15"CRT for MZ-5500/6500 | 124,000 | 59,800 |
| MZ-1D27 | MZ-25/28用デジタル/アナログディスプレイTV | 99,800 | 69,800 |
| MZ-1E01 | MZ-3500用RS-232Cボード | 28,000 | 13,000 |
| MZ-1E04 | MZ-2000用プリンタI/F | 10,000 | 6,000 |
| MZ-1E08 | MZ-2000/2200/80B用プリンタI/F | 9,000 | 8,000 |
| MZ-1E11 | MZ-6500用SFD I/F | 38,000 | 25,000 |
| MZ-1E14 | MZ-1500用クイックディスクI/F | 9,800 | 3,000 |
| MZ-1E18 | MZ-2000用クイックディスクI/F | 9,800 | 3,000 |
| MZ-1E21 | MZ-5500用GP I/F | 36,000 | 12,000 |
| MZ-1E22 | MZ-5500用GPIB I/F | 72,800 | 25,000 |
| MZ-1E29 | RS-232C I/F 300BT | 17,800 | 9,800 |
| MZ-1E32 | MZ-2500用パラレルI/F | 30,000 | 27,000 |
| MZ-1E33 | MZ-6500用パラレルI/F | 34,800 | 28,000 |
| MZ-1E44 | MZ-6500用S-RN I/F | 50,000 | 15,000 |
| MZ-1E45 | MZ-6500用RS-232C I/F | 50,000 | 15,000 |
| MZ-1M01 | MZ-2000/2200用16Bitボード | 78,000 | 8,000 |
| MZ-1M03 | MZ-5500用数値演算プロセッサ | 69,000 | 38,500 |
| MZ-1M09 | MZ-6500用8082-2演算プロセッサ | 82,000 | 30,000 |
| MZ-1M10 | MZ-2500パレットボード | 14,500 | 10,000 |
| MZ-1M12 | MZ-2861/6500/8087演算プロセッサ | 90,000 | 45,000 |
| MZ-1P06 | ドットプリンタ | 234,000 | 45,000 |
| MZ-1P10A | 24ドット80桁漢字プリンター | 245,000 | 79,000 |
| MZ-1P22 | 熱転写漢字プリンター | 59,800 | 19,800 |
| MZ-1P27 | 漢字水平プリンター | 268,000 | 98,000 |
| MZ-1P29 | 136桁水平漢字プリンター | 168,000 | 134,400 |
| MZ-1R01 | MZ-2000/2200Gボード | 39,800 | 10,000 |
| MZ-1R06 | MZ-5500用増設RAM | 45,000 | 8,000 |
| MZ-1R09 | MZ-5500VRAM | 35,000 | 15,000 |
| MZ-1R10 | MZ-5500漢字ROM付 | 30,000 | 9,800 |
| MZ-1R11 | MZ-5500用増設256KBRAM | 80,000 | 35,000 |
| MZ-1R12 | MZ-80B/2000/1500/700用RAM | 35,000 | 8,000 |
| MZ-1R14 | MZ-5500用辞書ROM | 40,000 | 22,000 |
| MZ-1R16 | MZ-5500用増設128KBRAM | 30,000 | 8,000 |
| MZ-1R21 | 漢字ROM MZ-IP10第二水準ROM | 38,000 | 13,000 |
| MZ-1R24 | MZ-1500用辞書ROM | 22,000 | 6,000 |
| MZ-1R26代 | MZ-2500用増設RAM | | 10,000 |
| MZ-1R32 | MZ-6500用RAM | 80,000 | 40,000 |
| MZ-1R35 | MZ-2861用1MB増設RAM | 55,000 | 19,000 |
| MZ-1R36 | MZ-2861用1MB増設RAM | 45,000 | 15,000 |
| MZ-1R37 | MZ-2500 640K RAMファイル | 34,800 | 37,000 |
| MZ-1S13 | MZ-1D17用チルトスタンド | 12,000 | 5,000 |
| MZ-1T02 | MZ-2200用テープレコーダー | 19,800 | 8,500 |
| MZ-1T03 | MZ-5500用テープレコーダー | 12,000 | 8,500 |
| MZ-1U09代 | MZ-2500用拡張ボード | | 7,200 |
| MZ-1V01 | パソコンプリンタ・コピー・ファクス | 278,000 | 85,000 |
| MZ-1X22 | MZ用モデムユニット | 21,800 | 13,000 |
| MZ-1X30 | MZ用1200/300モデムホン | 98,000 | 19,800 |
| MZ-2Z012 | MZ-5500付属ソフト | | 5,000 |
| MZ-2Z013 | MZ-5500 MS-DOS | 25,000 | 20,000 |
| MZ-2Z016 | MZ-5500付属ソフト | | 5,000 |
| MZ-2Z023 | MZ-5500 GW BASIC | 50,000 | 30,000 |
| MZ-2Z029 | MZ-6500 TODAY | 68,000 | 20,000 |
| MZ-2Z065 | MZ-6500書院日本語ワープロ | 69,800 | 28,000 |
| MZ-4Z001 | MZ-5500 IBM変換 | 30,000 | 8,000 |
| MZ-5511 | パソコン本体 | 288,000 | 35,000 |
| MZ-5521 | パソコン本体 | 388,000 | 55,000 |
| MZ-5Z013 | MZ-1500クイックディスク通信ソフト | | 3,500 |
| MZ-6F03 | クイックディスク | 450 | 400 |
| MZ-6P06 | MZ-IP06用トラクタフィーダ | 15,000 | 7,500 |
| MZ-6P18 | MZ-IP18/28用カットシートフィーダ | 60,000 | 35,000 |
| MZ-6P20 | MZ-IP22/17用ロールホルダー | 3,100 | 2,700 |
| MZ-6P27 | MZ-IP27用カットシートフィーダ | 58,000 | 39,800 |
| MZ-6P29 | MZ-IP29用カットシートフィーダ | 50,000 | 37,500 |
| MZ-6Z22 | MZ-6500用CP/M 86BASIC | 10,000 | 6,000 |
| MZ-6Z25 | MZ-50ストリーマユーティリティZプロセッサ | 39,800 | 15,000 |
| MZ-80P4B | 136桁ドットプリンタ | | 48,000 |
| MZ-80T20A | MZ-80用マシン語 | 6,000 | 5,000 |
| MZ-80T40A | MZ-80用PASCAL(言語) | 10,000 | 5,000 |
| MZ-80T70A | MZ-80用FDOS(OS) | 20,000 | 7,000 |
| MZ-80TU | MZ-80用システムプログラム | 20,000 | 8,000 |
| MZ-80TUB | MZ-80用バックアップツール | 20,000 | 8,000 |
| MZ-8376A | AXノートパソコン286N | 398,000 | 298,000 |
| MZ-8B104 | MZ-2000/2200用GPIB I/F | 45,000 | 18,000 |
| MZ-8BC01 | MZ-2000/2200用GPIB ケーブル | 18,000 | 8,000 |
| MZ-8BGK | MZ-80用 BGRAM2 | 39,000 | 10,000 |
| PA-7C18 | BASIC ICカード32KB | 15,000 | 13,500 |
| PA-7C19 | BASIC ICカード64KB | 25,000 | 22,500 |
| PA-9500 | 電子手帳 | 48,000 | 特価 |
| PA-9C90 | DB-Z RAMカード64KB | | 12,000 |
| PA-9C91 | DB-Z RAMカード128 | 20,000 | 17,000 |
| PC-1248DB | ポケコン10KB RAM | 11,000 | 9,800 |
| PC-1262 | ポケコン10.4KB RAM | 24,800 | 19,600 |
| PC-1280 | ポケコン8.4KB RAM | 24,800 | 19,600 |
| PC-1360 | ポケコン8KB RAM(カナ対応) | 29,800 | 19,800 |
| PC-1360K | ポケコン8KB RAM(漢字対応) | 36,800 | 32,800 |
| PC-1600K | ポケコン16KB RAM | 69,800 | 49,800 |
| PC-E200 | ポケコン32KB RAM | 22,000 | 17,800 |
| PC-E500 | ポケコン32KB RAM | 28,800 | 19,800 |
| PC-E500BL | ポケコン32KB RAM | 28,800 | 19,500 |
| PC-E550 | ポケコン64KB RAM | 32,000 | 特価 |
| SII-6BE1-1M | I/OデーターCZ-500C増設RAM | 25,000 | 20,000 |
| TP-18 | カセットデーターレコーダー（ ポケコン用） | 10,800 | 9,800 |
| UE-01 AX | ICカードインターフェイス | 45,000 | 30,000 |
| UE-1E02 AX | ICカードインターフェイス | 45,000 | 30,000 |
| UE-1R13 AX | 1M増設RAMボード | 100,000 | 65,000 |
| UE-1R07 AX | 辞書ROMボード | 32,800 | 26,200 |
| UE-1R09 AX | 1M増設RAMボード | 75,000 | 52,500 |
| UE-1R03 AX | 辞書ROMボード | 32,800 | 25,000 |
| UE-1U01 AX | スロットボックス | 5,000 | 4,000 |
| UN-X68 | キーボードシリコンカバー(PRO) | 3,500 | 2,800 |
| UX-1 | ホームコピーファクス | 78,000 | 69,800 |
| XC-100P | イメージプレゼンテーター長ケーブル15P(3m) | 398,000 | 298,000 |
| XC-10SC1 | 24K TO 15Kスキャンコンバータ | 300,000 | 240,000 |
| IP-1246B | パソコンファクス98 | 98,000 | 78,000 |
| IP-1243 | パソコンファクス25 | 30,000 | 8,000 |
| X1-将軍 | X1 ワープロ 5"2HD.2D | 34,800 | 29,000 |
| X1-待46B | X1 ワープロ 5".2D2D | 19,800 | 16,800 |
| MACRO-80 | X1 マクロアッセンブラ5"2D | 20,000 | 17,500 |
| MACRO-80 | MM-7/77/8801/8001 マクロアッセンブラ 5"2D | 20,000 | 17,500 |

※富士通、NEC、シャープ周辺機器(拡張機器全機種、プリンター他)も常時取り扱っております。

**〈全商品新品完全保証付〉**

シャープ、カシオポケコン全機種取扱い。カタログ、価格表ご請求には、72円 切手を添えてお願い致します。

通信販売のお問い合せ、御注文は

# 0426-45-3001(本店)

**FAX.0426-44-6002**

●営業時間/10:00~19:00●電話受付/20:00迄可●定休日/水曜日

**SHARP SUPER EXE SHOP**

アイビット電子株式会社 〒192 東京都八王子市北野町560-5

16号バイパス
1Fショップ
2Fパソコン教室
ココ(ご来店は駅前徒歩0分)
至京王八王子
京王北野駅
至新宿
至横浜

上記の広告商品はすべて店頭販売もしております。

**全国通信販売 北海道から沖縄まで**

★送料はご注文の際にお問い合わせ下さい。
★掲載の商品は、すべて新品、保証書付きです。
★掲載の商品は充分用意してありますが、ご注文の際は、在庫の確認の上、現金書留または、銀行振込でお申し込み下さい。全商品クレジットでも扱っております。
★お申し込みの際は必ず電話番号を明記して下さい。
★商品、品切れの節はご容赦下さい。

富士銀行八王子支店 (普)1752505

●本誌発売時には上記価格よりさらにお求めやすい価格に変更されている場合があります。●この広告の商品にはすべて送料・消費税は含まれておりません。

X68000 Pro SHOP

# BASIC HOUSE

TEL0286-22-9811/FAX0286-25-3970

# 年末年始大特価

## X68000XVI/SUPER/PROII―PROSHOPならではのサポート&価格をお届けします。

XVI/SUPERお買上で
おすきなゲーム1本
X68000ロゴバッヂ
X68000ディスクラベル
2HDdisk1箱
プレゼント!

※ソフトプレゼントは定価¥10,000以下の物とさせていただきます。

### XVI(HD100)
買替組へ100MHDD内蔵XVI

定価¥518,000
Basic House特価 ¥398,000

| 12回 | 24回 | 36回 | 48回 |
|---|---|---|---|
| ¥36,200 | ¥19,200 | ¥13,300 | ¥10,500 |

### SUPER(HD100)
B.H.オリジナル100MHDD内蔵

定価¥597,800
Basic House特価 ¥269,000

| 12回 | 24回 | 36回 | 48回 |
|---|---|---|---|
| ¥24,200 | ¥12,800 | ¥8,900 | ¥7,000 |

### 今月の超目玉! X68000PROII & CZ-606D

限定 早い者勝ち!!

¥198,000

| 12回 | 24回 | 36回 | 48回 |
|---|---|---|---|
| ¥17,600 | ¥9,300 | ¥6,500 | ¥5,100 |

### XVI&CZ-614D
0.31ピッチ、3モードスキャンCRT/TVset

定価¥503,000
Basic House特価 ~~¥418,000~~

| 12回 | 24回 | 36回 | 48回 |
|---|---|---|---|
| ¥36,200 | ¥19,200 | ¥13,300 | ¥10,500 |

### SUPER&CZ-606D
0.31ピッチ、2モードスキャンCRT/TVset

定価¥427,800
Basic House特価 ~~¥298,000~~

| 12回 | 24回 | 36回 | 48回 |
|---|---|---|---|
| ¥24,200 | ¥12,800 | ¥8,900 | ¥6,200 |

### XVI(HD100)&CZ-614D
B.H.オリジナル100MHDD内蔵XVI

定価¥653,000
Basic House特価 ¥498,000

| 12回 | 24回 | 36回 | 48回 |
|---|---|---|---|
| ¥45,000 | ¥23,900 | ¥16,500 | ¥13,000 |

### SUPER(HD100)&CZ-606D
B.H.オリジナル100MHDD内蔵

定価¥577,800
Basic House特価 ¥328,000

| 12回 | 24回 | 36回 | 48回 |
|---|---|---|---|
| ¥29,500 | ¥15,700 | ¥10,800 | ¥8,500 |

### 増設メモリ&コプロセッサボード

KGB-X68PRKIIシリーズ
ご購入後のメモリの増設が可能です。

| | | |
|---|---|---|
| 2M実装/コプロ別売り | PRKII-02 | ¥ 55,000 |
| 4M実装/コプロ別売り | PRKII-04 | ¥ 90,000 |
| 6M実装/コプロ別売り | PRKII-06 | ¥125,000 |
| 8M実装/コプロ別売り | PRKII-08 | ¥160,000 |
| 2M実装/コプロ付属 | PRKII-12 | ¥ 85,000 |
| 4M実装/コプロ付属 | PRKII-14 | ¥120,000 |
| 6M実装/コプロ付属 | PRKII-16 | ¥155,000 |
| 8M実装/コプロ付属 | PRKII-18 | ¥190,000 |

### 旧PRK処分特価

PRKIIの新発売に伴い、旧型PRKを大特価販売いたします。
在庫分のみですので品切れの際にはご容赦ください。

TEL CALL!!

### FHD-200

計側技研オリジナル
SCSIハードディスク
X68000/Mac対応

定価¥298,000
Basic House特価 ¥203,000

### Infinity40turbo

PLI社リムーバブル42MB
SCSIハードディスク
X68000/Mac対応/メディア2枚サービス

Basic House特価 ¥148,000
メディア2枚&SCSIインターフェースセット
Basic House特価 ¥170,000

### A4フルカラースキャナ

2台限り!
SHARP JX-220
(CZ-8NS1コンパチ)

定価¥148,000
Basic House特価 ¥95,000

*パラレルインターフェースは付属しません。

### PRINTER

SHARP CZ-8PC5
定価¥96,800
Basic House特価 ¥82,000

EPSON AP-1000
定価¥97,800
Basic House特価 ¥81,000

※プリンタお買い上げの方にもれなくカラーインクリボン&熱転写用紙をプレゼント。

### ビデオボード&ケース

CZ-6BVI&Basic House製ケース(KGB-BVBX)の特別セット

定価合計¥30,800
Basic House特価 ¥25,600

### マウス大特価

CZ-8NM3
Basic House特価 ¥7,800

CZ-8MN2A
Basic House特価 ¥5,400

CZ-8NT1
Basic House特価 ¥11,000

### ローランドMIDI音源

| | |
|---|---|
| CM-32L | ~~¥ 69,000~~ |
| CM-64 | ~~¥129,000~~ |
| CM-300(NEW) | ~~¥ 58,000~~ |
| CM-500(NEW) | ~~¥115,000~~ |

### BF-68PRO

CZシリーズ用CRTフィルタ
定価¥19,800
Basic House特価 ¥16,200

※CRTとセットなら、さらにお安く致します。

### MODEM

omron MD-24FB5V
NEC COMSTARZ CLUB 24/5

定価¥39,800
Basic House特価 ¥32,800

### Basic Houseオリジナルハードウェア

#### for X68000

| | |
|---|---|
| 12bit, 8/16ch 高速A/Dコンバータ (Xbasic, XC, アセンブラ用ライブラリ付属) | ¥128,000 |
| 12bit 4-16ch高速D/Aコンバータ (Xbasic, XC, アセンブラ用ライブラリ付属) | 発売予定 |
| 16bit絶縁型、パラレルインターフェース (Xbasic, XC, アセンブラ用ライブラリ付属) | ¥ 68,000 |
| 64180CPUアドオンボード(Mach180) (専用インターフェース, CP/M80エミュレータ付属) | ¥ 98,000 |
| ハンディプリンタ(Handy Print Jack) | ¥ 24,800 |
| ユニバーサルボード | ¥ 6,800 |
| ビデオボードケース (CZ-6BV1を外付けにします。) | ¥ 9,800 |

#### for X1/turbo

| | |
|---|---|
| 12bit 16ch、高速A/Dコンバータ | ¥118,000 |
| 12bit 4ch、高速D/Aコンバータ | ¥ 98,000 |
| 16bit絶縁型パラレルインターフェース | ¥ 42,000 |
| GPIBインターフェース | ¥ 58,000 |
| 汎用8bitA/D&24bit(TTL)パラレルI/O | ¥ 19,800 |
| ハードディスクインターフェース(turbo専用) | ¥ 16,000 |

### Basic Houseオリジナルソフトウエア

| | |
|---|---|
| BASIC拡張関数パッケージ (Xbasicの外部関数集) | ¥ 9,800 |
| C言語ライブラリ (拡張関数パッケージのBas ToC用ライブラリ) | ¥ 6,800 |
| BASIC拡張関数パッケージ C言語ライブラリ付 | ¥ 14,800 |
| ディスクキャッシャー (SASI HDDとFDDのアクセスを高速化出来ます。) | ¥ 6,800 |
| CP/M68Kエミュレータ (Human68K上でCP/M68Kのコマンドを実行できます。) | ¥ 19,800 |

低金利クレジット・通信販売料 全国一律¥1,000・長期クレジット可能　※表示価格に消費税は含まれておりません。

株式会社 計測技研/マイコンショップBASIC HOUSE 〒321 栃木県宇都宮市竹林町503-1

価格に自信あり!!

## OAB特選〜X68000シリーズセット

★本体・ディスプレイセットでお買い上げの方にはゲームソフト2本付

### ①X68000XVI
- CZ-634C-TN
- CZ-614D-TN
- MD-2HD 20枚

定価合計¥503,000
特価
¥TEL下さい!!

●SX-WINDOW搭載!!

### ②X68000XVI-HD
- CZ-644C-TN
- CZ-614D-TN
- MD-2HD 20枚

定価合計¥653,000
特価
¥TEL下さい!!

☆本体、モニターのみの方は、さらにお安くなります。

★X68000XVIお買い上げの方にもれなく数値演算プロセッサー(MB68881RC16)をプレゼントします!!

### ③X68000 PROII
●SX-WINDOW搭載!!
- CZ-653C-BK/GY
- CZ-606D-BK/GY
- MD-2HD 20枚

定価合計¥364,800

特価¥218,000

### ④X68000PRO II-HD
- CZ-663C-BK/GY
- CZ-605D-BK/GY
- MD-2HD 20枚

定価合計¥510,000

安くて表示できません。

### X68000 SUPER-HD
- SX-WINDOW搭載
- SCSI I/F装備
- 80MBハードディスク搭載
- 3MB大容量メモリ装備
- 高解像度グラフィック

●SX-WINDOW搭載!!

価格応談
表示よりも安く

### ⑤X68000 SUPER-HD
- CZ-623C-TN(チタン)
- CZ-614D-TN(チタン)
- MD-2HD 20枚

定価合計¥633,000
特価¥366,000

### X68000 特選OABセット ★本体のみ単品OK!!

価格応談
表示価格よりも安く

① CZ-604C-TN(新品)+CZ-606D-TN(新品)
3セット限り………特価¥268,000

② CZ-604C-TN(新品)+CZ-614D-TN(新品)
1セット限り………特価¥306,000

③ CZ-603C-BK(新品)+CZ-603D-BK(新同品)
3セット限り………特価¥218,000

④ CZ-612C-BK(新品)+CZ-606D-BK(新同品)
2セット限り………特価¥227,000

ビッグな商品手ごろな価格!

# オーエーブレイン 全国通販

業社販売致します!! お問合せは、杉山宛迄。

★全商品保証書付。専門のアドバイザーがお客様のニーズに親切に対応します。
★初期不良・輸送トラブル等に迅速に対応し、即交換させていただきます。
★送料は、着払いとなります。

●ご注文、お問合せは…毎日午前10時から午後8時まで
●下取・買取は電話で見積りしております。責任を持って下取りさせて頂きます。
●商品のお届けは…入金確認後、即日発送致します。

## 周辺機器コーナー

### プリンターセットコーナー

- CZ-6PVI(カラービデオプリンター) 定価¥198,000……▶特価¥147,800
- CZ-8PC3(24ドット熱転写カラープリンター) 定価¥65,800……▶特価¥52,800
- CZ-8PK10(24ピン漢字ドットプリンター・136桁) 定価¥97,800……▶特価¥70,800
- CZ-8PGI(24ピンカラー漢字ドットプリンター・80桁) 定価¥130,000……▶特価¥91,800
- CZ-8PG2(24ピンカラー漢字ドットプリンター・136桁) 定価¥160,000……▶特価¥113,800
- IO-735XB(カラーイメージェットプリンター) 定価¥248,000……▶特価¥169,000

### X68000用ソフトウェアー・コーナー

| | 定価 | 特価 |
|---|---|---|
| ①CZ-212BS(BUSINESS) | ¥68,000 | ¥53,000 |
| ②CZ-220BS(DATA) | ¥58,000 | ¥45,000 |
| ③CZ-215MS(Sampling) | ¥17,800 | ¥13,800 |
| ④CZ-221HS(NEW Print Shop) | ¥10,800 | ¥15,500 |
| ⑤CZ-227BS(TOP財務会計) | ¥200,000 | ¥158,000 |
| ⑥CZ-226BS(CARD) | ¥229,800 | ¥23,000 |
| ⑦CZ-223CS(Communcabion) | ¥19,800 | ¥115,500 |
| ⑧CZ-213MS(MUSIC) | ¥18,800 | ¥14,800 |
| ⑨CZ-211LS(C compiler) | ¥39,800 | ¥31,000 |
| ⑩C-TRACE(キャスト) | ¥68,000 | ¥52,000 |
| ⑪EW(イースト) | ¥38,000 | ¥29,000 |

### 特選品

■CZ-8PC5(48ドット熱転写カラー漢字プリンター)(定価¥96,800)安くて表示できません。

## X68000用周辺機器コーナー

| 型番 | 品名 | 定価 | 特価 |
|---|---|---|---|
| CZ-6BEI | IBM増設RAMボード | ¥35,000 | ¥25,200 |
| CZ-6BEIB | IBM増設RAMボード | ¥28,000 | ¥20,200 |
| CZ-6BE2 | 2MB増設RAMボード | ¥79,800 | ¥58,700 |
| CZ-6BE4 | 4MB増設RAMボード | ¥138,000 | ¥102,200 |
| CZ-6BF1 | 増設用RS-232Cボード | ¥49,800 | ¥36,700 |
| CZ-6BGI | GP-IBボード | ¥59,800 | ¥43,200 |
| CZ-6BMI | MIDIボード | ¥26,800 | ¥19,200 |
| CZ-6BNI | スキャナ用パラレルボード | ¥29,800 | ¥21,700 |
| CZ-6BPI | 数値演算プロセッサボード | ¥79,800 | ¥58,700 |
| CZ-6BOI | ユニバーサルI/Oボード | ¥39,800 | ¥29,200 |
| CZ-6EBI/BK | 拡張I/Oボックス | ¥88,000 | ¥63,700 |
| CZ-6VTI/BK | カラー・イメージ・ユニット | ¥69,800 | ¥50,700 |
| CZ-8NM24 | マウス | ¥6,800 | ¥4,700 |
| CZ-8NTI | マウストラックボール | ¥9,800 | ¥6,700 |
| CZ-8NSI | カラーイメージスキャナー | ¥188,000 | ¥134,700 |
| CZ-6BCI | FAXボード | ¥79,800 | ¥58,700 |
| CZ-8TM2 | モデムユニット | ¥49,800 | ¥36,700 |
| CZ-64H | 増設ハードディスク | ¥120,000 | ¥86,700 |
| CZ-6TU GY/BK | RGBシステムチューナー | ¥33,100 | ¥23,700 |
| BF-68PRO | 高性能CRTフィルタ | ¥19,800 | ¥14,700 |
| CZ-6MOI | 光磁気ディスクユニット | ¥450,000 | ¥326,700 |
| CZ-6BSI | SCSIインターフェースボード | ¥29,800 | ¥21,700 |
| CZ-6BL2 | LANボード | ¥298,000 | ¥216,700 |

## I・O DATA 増設RAMボード 限定

●1MB増設PAMボード PIO-6BEI-A
定価¥25,000

特価¥16,800

●2MB増設RAMボード PIO-6BE2-2M
定価¥50,000

特価¥33,300

●4MB増設RAMボード PIO-6BE4-4M
定価¥88,000

特価¥58,300

## ハードディスク

★その他特価品有!TEL下さい!!

■シャープ CZ-64H……特価¥86,000
CZ-68H……特価¥118,000
■ロジテック LHD-200……特価¥218,000
■アイテム HXD-040……特価¥88,000
HXD-042……特価¥95,000

■アイテック 価格応談
●TX-80……特価¥~~77,800~~
●TX-130……特価¥~~91,000~~
●TX-180……特価¥~~122,000~~
★SCSIボード……特価¥22,000

## オーエーブレイン今月の特価品!! 台数限定 お早目に!!

- KGB-X68PRKII-02(¥55,000)…特価¥42,800
- PRKII-04(¥90,000)…特価¥70,200
- PRKII-06(¥125,000)…特価¥97,500
- PRKII-08(¥160,000)…特価¥124,800
- PRKII-12(¥85,000)…特価¥66,300
- PRKII-14(¥120,000)…特価¥93,600
- PRKII-16(¥155,000)…特価¥121,000
- PRKII-18(¥190,000)…特価¥148,000
- MC-6888 IRC (¥38,000)…特価¥28,500

〈開発ツール〉●C-コンパイラPRO68KV.2
定価¥44,800 CZ-245IS ……特価¥33,000

〈C言語〉●C&Professional Pack
定価¥58,000 ……特価¥40,500

〈データベース〉●CARD PRO68K Ver.2.0
定価¥29,800 CZ-253BS ……特価¥21,000

〈音楽〉●Music studio PRO68K Ver.2.0
定価¥28,800 CZ-261MS ……特価¥21,300

〈CGシール〉●CANVAS PRO68K
定価¥29,800 CZ-249GS ……特価¥22,200

〈通信〉●Tlepotion PRO68K
定価¥22,800 CZ-258BS ……特価¥17,00

〈ワープロ〉●Multiword PRO68K
定価¥32,000 CZ-225BS ……特価¥23,80

〈グラフィック〉●Z's STAFF PRO68K Ver.2.
(シャフト)定価¥58,000 ……特価¥38,50

〈グラフィック〉●C-TRACE68 Ver.3.0
定価¥98,000 ……特価¥69,00

## 中古パソコン

| 機種 | 価格 |
|---|---|
| PC-9801RA2 | ¥198,000より |
| PC-9801RX2 | ¥130,000より |
| PC-9801VX21 | ¥125,000より |
| PC-9801VM21 | ¥90,000より |
| PC-9801VM2 | ¥75,000より |
| PC-9801F2 | ¥18,000より |
| PC-9801EX2 | ¥130,000より |
| PC-9801UV21 | ¥65,000より |
| PC-9801LV21 | ¥93,000より |
| PC-286V | ¥75,000より |
| PC-286VE | ¥80,000より |
| PC-286L | ¥80,000より |
| PC-286LS | ¥190,000より |
| PC-8801FH | ¥18,000より |
| PC-8801MA | ¥25,000より |
| X68000 | ¥110,000より |
| X68000(HD) | ¥160,000より |
| X1ターボZII | ¥28,000より |
| FM77AV40EX | ¥15,000より |
| 200ラインCRT | ¥8,000より |
| 400ラインCRT | ¥30,000より |
| 80桁プリンタ | ¥15,000より |
| 135桁プリンタ | ¥35,000より |

●その他多数有り、お問い合せ下さい。

## ユニット

- FD-1155D(5インチ)……¥9,000.
- FD-1155C(5インチ)……¥8,000.
- FD-1165A(8インチ)……¥3,000.
- FD-1137D(3.5インチ)……¥9,000.
- D-5146H(5インチ40MB)……¥29,000.
- D-3142(3.5インチ40MB)……¥29,000.
- D-3148(3.5インチSISC)……¥30,000.
- 外付8インチ2ドライブ……¥20,000.
- 外付5インチ2ドライブ……¥30,000.

## 通信販売によるご購入方法(お電話でお申し込み下さい。)

### 現金一括払い
銀行振込:電信扱いにてお振込下さい。
手数料はお客様負担となります。
現金書留:住所、氏名、電話番号、商品名、使用機種、メディア等をお書き添えのうえ、現金書留にて当社までお送り下さい。

### クレジット
専用のお申し込み用紙をお送り致しますので、必要事項をご記入・捺印のうえ、ご返送下さい。
※未成年者の方は、保護者のご承認を受けてからお申し込み下さい。

### 振込先
●第一勧業銀行 御徒町支店 (普)1376679 オーエーブレイン
●朝日信用金庫 本店 (普)334833 オーエーブレイン

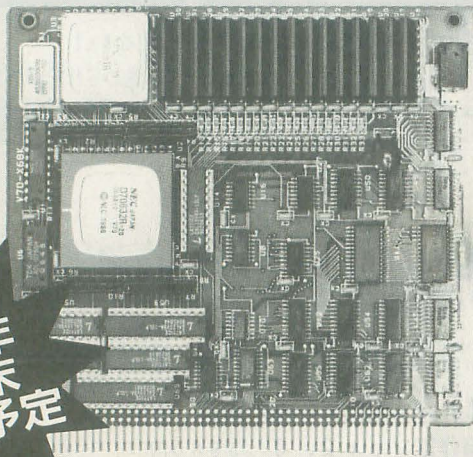
★クレジットは1〜60回払いで月々5,000円よりご自由に設定できます。

オーエーブレイン 〒110 東京都台東区台東1-28-4 TEL & FAX 5688-3621

■流通事情により、広告表示よりお安くなる場合もございます。まずは、お電話下さい。■ビジネス・ゲームセットもございます。

ACCESS

# あなたの X68000が Super ワークステーションに！

## V70 +AFPP アクセラレータ

**VDTK-X68K**(V70 Development Tool Kit-X68K)

- V70 CPU 20MHz(μPD70632)
  32ビットマイクロプロセッサ
- V70 AFPP(μPD72691)
  フローティング・ポイント・プロセッサ
- メインメモリ DRAM 2MByte
  同一ページ内アクセスは No Wait
- 共有メモリ SRAM 128KByte
  X68000との通信用

**同梱ソフト**

- アセンブラ
- ソースコードデバッガ
- システムモニタ
- フローートエミュレータ

**オプションソフト(Cコンパイラ)**
VDTK-C-X68K(V70 Development Tool Kit-C-X68K)

## フローティング・ポイント・プロセッサ AFPP標準搭載

## 最速10MIPS※1 5.8M FLOPS※2

## 無限の可能性を秘め この冬 X68000上に堂々デビュー

## V70のあらたなるパワーをあなたの手に！

本製品はX68000上でV70のアプリケーションを開発して頂くための開発キットです。

V70で記述されたプログラムの動作環境を提供するシステムモニタを用意しておりますのでX68000のIOCSや、Human68kと同等のシステムコールを利用して、容易にプログラムを実行させることが可能です。また、V70をサブCPUとして浮動小数点演算を行わせる、X68000用デバイスドライバも用意しております。これにより高速な浮動小数点演算が可能となります。

さらにV70CPUに加えAFPPを標準で搭載しております。20MHzの高速クロックで使用しており、より高速な数値演算が可能です。32ビットマイクロプロセッサV70の特徴である仮想記憶、メモリプロテクション、CPUレベルでのデバッグ機能などをサポートし、効率の良い開発環境を提供いたします。

| | |
|---|---|
| VDTK-X68K | ¥248,000 |
| VDTK-C-X68K | ¥68,000 |

**購入方法**

上記商品は当面の間、通信販売のみとさせて頂きます。
購入希望の方は住所、(社名、所属)氏名、電話番号をお知らせ下さい。注文書をお送りいたします。

※1 V70 レジスターレジスタ間基本命令、NOP命令(実測)
※2 AFPP ベクトル／行列演算(倍精度)μPD72691ユーザーズマニュアルより
※本製品は、有限会社アクセスと株式会社ハドソンの共同開発製品です。


有限会社アクセス 〒101 東京都千代田区神田神保町1-64 神保町協和ビル7F
☎03(3233)0200(代) FAX.03(3291)7019

# パソコン/ワープロ通信ネットワークサービス
# J&P HOTLINE

**J&P HOTLINEは パソコン通信の原点。**

## 永吉 孝夫さん 36歳
**(JH269054 つちのこ ふたり) 弁護士**

大阪の中之島、市役所や裁判所などの官庁が並ぶ一角に永吉さんの事務所はあります。弁護士さんという職業を聞いていささか緊張して取材に伺いましたが、ドアをノックすると中から「どーーぞー」と、サスガニ!?大きい声。
細やかな心配りとおおらかさを兼ね備えた、パソコン大好き先生という感じ。自作のゲームを前にしたときの満面の笑みが印象的でした。

## ハードな仕事の息抜きには、「パソコン」が一番。

弁護士といえば、あこがれる人も多い花形職業。でももちろん仕事の中身はハードで、頭脳はもちろん、精神も身体も緊張の連続。
永吉さんとコンピュータの出会いは、10年前に遡り、PC-1211というシャープのポケットコンピュータを購入されたことから始まります。当時はBASICのゲームに夢中だったとか。その後、ワープロが200万円もするような初期の時代に職場の上司を説得して購入させたり、ご自身も独立されてワープロやパソコンを数台買い換えられました。でも、お仕事で使うというより、緊張をリラックスさせるという使い方の方が多いそうです。J&P HOTLINEとの出会いも早く、パソコン通信というものが注目され始めて、一般の雑誌等でクローズアップされた頃、行きつけのJ&Pのお店でスタータキットを購入して会員に。'88.11.8にSIG「ディオニュソス」に初めての書き込み。その文章をちゃんと保管しておられました。「J&P HOTLINEは、パソコン通信に第一歩を踏み出した記念すべきネットですからね」と。
永吉さんがパソコンの前に座っておられる時間は、かなり長いようですが、全てが遊びや息抜きという訳ではありません。「大阪弁護士協同組合」にホストを持つ「Ben-Ben Net」という草の根ネットの中心メンバーとして大活躍。同組合の情報処理委員会の一員である永吉さんは、判例フォローや裁判官情報・法務局の住所変更等、データベースの構築や、SIGのパワーライターとして、時にはみんなの息抜きのために自作ゲームのアップロードなど、フル回転といったところです。とにかく「何でもやってみよう」と、チャレンジ精神旺盛な永吉さん。J&P HOTLINEでも、今後ますますの活躍が期待されます。

**超多忙な永吉さんには、HOTLINEの温かい雰囲気が心地よい。**

★初めて書き込みをしたSIG「ディオニュソス」は、今でもお気に入り。最近はROMっ子専門だそうですが、じっくりとながーいお付き合いをしていきたいとのこと。
★Ben-Ben Netの参考としても、HOTLINEは大活躍。
★自作プログラムには、豊富なプログラムライブラリーを活用。
★コンピュータミュージックやグラフィックにも興味を持って、こつこつと手掛けているので、そのうち情報交換・作品交換もしたい……と、夢は膨らみます。

**J&P HOTLINEへのご入会はスタータキットで。**

お求めは、下記のお店へ。又は現金書留にて、¥3,000+¥90(消費税3%)=¥3,090を事務局までお送り下さい。
すぐにスタータキットをお送りします。

お問い合わせは 〒556 大阪市浪速区日本橋西1-6-5 上新電機株式会社
J&P HOTLINE事務局宛 TEL.(06)632-2521

**スタータキットのお求めはJ&P各店でどうぞ。**

| 店名 | 住所 | 電話 |
|---|---|---|
| 渋谷店 | 東京都渋谷区道玄坂2丁目28番4号 | ☎(03)3496-4141 |
| 町田店 | 東京都町田市森野1丁目39番16号 | ☎(0427)23-1313 |
| 八王子店 | 東京都八王子市旭町1番1号八王子そごう7F | ☎(0426)26-4141 |
| 立川店 | 東京都立川市幸町4-39-1 | ☎(0425)36-4141 |
| 本厚木店 | 厚木市中町3-4-3 | ☎(0462)25-1548 |
| 富山店 | 富山市掛尾町300番地 | ☎(0764)22-5033 |
| 金沢店 | 金沢市入江2-63 | ☎(0762)91-1130 |
| 寺地店 | 金沢市寺地2-3 | ☎(0762)47-2524 |
| 大須店 | 名古屋市中区大須4丁目2-48 | ☎(052)262-1141 |
| テクノランド | 大阪市浪速区日本橋5丁目6番7号 | ☎(06) 634-1211 |
| メディアランド | 大阪市浪速区日本橋5丁目8番26号 | ☎(06) 634-1511 |
| コスモランド | 大阪市浪速区難波中2丁目1番17号 | ☎(06) 634-3111 |
| U. S. LAND | 大阪市浪速区日本橋4丁目9番15号 | ☎(06) 634-1411 |
| ビジネスランド | 大阪市北区梅田1-1-3大阪駅前第3ビルB2 | ☎(06) 348-1881 |
| 梅田店 | 大阪市北区小松原町1-10 | ☎(06) 362-1141 |
| 高槻店 | 高槻市高槻町11番16号 | ☎(0726)85-1212 |
| くずは店 | 枚方市楠葉花園町15番2号 | ☎(0720)56-8181 |
| 千里中央店 | 豊中市新千里東町1-3 SENCHU PAL 2番街4F | ☎(06) 834-4141 |
| 摂津富田店 | 高槻市大畑町24-10 | ☎(0726)93-7521 |
| 寝屋川店 | 寝屋川市緑町4-20 | ☎(0720)34-1166 |
| 枚方バイパス店 | 枚方市田口3-41-7 | ☎(0720)48-1211 |
| 藤井寺店 | 藤井寺市岡2丁目1番33号 | ☎(0729)38-2111 |
| 岸和田店 | 岸和田市土生町2451-3 | ☎(0724)37-1021 |
| さんのみや1ばん館 | 神戸市中央区八幡通3-2-16 | ☎(078)231-2111 |
| 西宮店 | 兵庫県西宮市河原町5-11 | ☎(0798)71-1171 |
| 伊丹店 | 伊丹市昆陽池1-63 | ☎(0727)77-5101 |
| 姫路店 | 姫路市東延末1丁目1番住友生命姫路南ビル1F | ☎(0792)22-1221 |
| 京都寺町店 | 京都市下京区寺町通仏光寺下ル恵比須之町549 | ☎(075)341-3571 |
| 京都近鉄店 | 京都市下京区烏丸通七条下ル東塩小路町702 | ☎(075)341-5769 |
| 和歌山店 | 和歌山市元寺町4丁目4番地 | ☎(0734)28-1441 |
| 奈良1ばん館 | 奈良市三条町478-1 | ☎(0742)27-1111 |
| 郡山インター店 | 大和郡山市横田693-1 | ☎(07435)9-2221 |
| 熊本店 | 熊本市手取本町4-12 | ☎(096)359-7800 |

T1002179010605 雑誌 02179-1